WHOI-93-25

An Experimental Study of Air Entrainment by Breaking Waves

by

Eric Lamarre

Woods Hole Oceanographic Institution
Woods Hole, Massachusetts 02543

and

The Massachusetts Institute of Technology
Cambridge, Massachusetts 02139

DTIC
S ELECTE D
JUL 2 1 1995
G

June 1993

DOCTORAL DISSERTATION

Funding was provided by The Office of Naval Research and the National Science Foundation through the Massachusetts Institute of Technology.

Reproduction in whole or in part is permitted for any purpose of the United States Government. This thesis should be cited as: Eric Lamarre, 1993. An Experimental Study of Air Entrainment by Breaking Waves. Ph.D. Thesis. MIT/WHOI, WHOI-93-25.

Approved for publication; distribution unlimited.

Approved for Distribution:

George V. Frisk

George V. Frisk, Chair
Department of Applied Ocean Physics and Engineering

[illegible]

John W. Farrington
Dean of Graduate Studies

19950720 017

AN EXPERIMENTAL STUDY OF AIR ENTRAINMENT BY BREAKING WAVES

by

ERIC LAMARRE

B.Eng. Civil Engineering and Applied Mechanics
McGill University (1988)

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF

DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

and the

WOODS HOLE OCEANOGRAPHIC INSTITUTION

May 1993

© Massachusetts Institute of Technology and
Woods Hole Oceanographic Institution, 1993

Accesion For		
NTIS CRA&I		☒
DTIC TAB		☐
Unannounced		☐
Justification		
By		
Distribution /		
Availability Codes		
Dist	Avail and / or Special	
A-1		

Signature of Author [illegible]
Joint Program in Applied Ocean Science and Engineering
Massachusetts Institute of Technology
Woods Hole Oceanographic Institution

Certified by [illegible]
Professor W. Kendall Melville
Visiting Professor, Civil & Environmental Engineering
Thesis Supervisor

Accepted by [illegible]
Professor Arthur B. Baggeroer, Chairman
Joint Program in Applied Ocean Science and Engineering

AN EXPERIMENTAL STUDY OF AIR ENTRAINMENT BY BREAKING WAVES

by

ERIC LAMARRE

Submitted in partial fulfillement of the requirements
for the degree of DOCTOR OF PHILOSOPHY at the
Massachusetts Institute of Technology
and the
Woods Hole Oceanographic Institution
May 15, 1993

ABSTRACT

Breaking waves charge the surface layer of the ocean with small air bubbles which play an important role in air-sea gas transfer and in underwater acoustics near the ocean surface. This work reports on a series of laboratory and field experiments on the measurement on air entrainment by breaking waves.

The first part of this thesis addresses the measurement of high volumetric concentrations of air (0.3% to 100% void-fraction) found immediately beneath breaking waves. Instrumentation based on the change of electrical impedance of the bubbly mixture with change in void-fraction is developed. Laboratory measurements are conducted in a wave channel and in a large three-dimensional wave basin. Maps of the evolution of the void-fraction distribution in bubble plumes generated by various size breaking waves are presented. Moments of the void-fraction field are shown to scale with the initially enclosed air volume at breaking and the energy dissipated by breaking. A significant fraction (30 to 50%) of the energy dissipated by breaking is found to be expanded in entraining bubbles against their buoyancy. The results reveal that the bubble plumes experience rapid transformations within the first wave period after the onset of breaking. In particular, the plumes loose 95% of the initially entrained air volume during the first wave period. Predictions of the low-frequency resonant oscillations of the bubble plumes from measurements of the void-fraction compare well with acoustic measurements. Measurements near the ocean surface show high void-fractions up to 24% immediately beneath breaking waves. These are several orders of magnitude greater than previously reported time-averaged measurements.

The second part addresses the measurement of very low void-fractions. Instrumentation based on the propagation velocity of low-frequency acoustic pulses is developed. Simultaneous measurements of the sound-speed (and thus the void-fraction) at several depths are conducted during two field experiments. Time-series of sound-speed and attenuation show dramatic fluctuations over time periods on the order of minutes or less. These are attributed to the formation of bubble plumes or passage of bubble clouds. Frequent occurences of sound-speed anomalies greater than 100m/s and attenuation greater than 30dB/m are observed for moderate wind conditions (8m/s). The signals at various depths are highly correlated and mostly coherent at frequencies below 0.05Hz. The time-averaged (20min) sound-speed profile is found to be significantly more pronounced and shallower than previously reported. Simultaneous measurements at several acoustic frequencies show that the sound-speed is non-dispersive below 20kHz for moderate wind conditions. Bubble size distributions are inferred from the sound-speed and attenuation measurements.

Thesis Supervisor: Professor W. Kendall Melville

Title: Visiting Professor of Civil & Environmental Engineering

A mes parents, Louise et François Lamarre

Acknowledgments

My thesis advisor Ken Melville, has provided the support and the guidance for this work. He has contributed many of the ideas presented here and his involvement was significant at all levels of this research. His trust and confidence in my abilities and his enthusiasm for experimental work were the main reasons for my enrolling in the Ph.D. program. He deserves the credit for having made my research experience at MIT so enjoyable and fulfilling. Thanks Ken!

I also want to thank the other members of my thesis committee, Professor Hemond, Professor Kytömaa and Dr. Williams III. Thanks to Harri Kytömaa for helpful advice in the developmental stage of the void-fraction instrument and thanks to Sandy Williams for helping me with the sound-speed instrumentation. Sandy taught me much about the many crucial details involved in acoustic field experiments.

Laboratory and field experiments typically require the help of many committed people and mine have been no exception. My colleague Andy Jessup taught me about the art of conducting field experiments. He helped me in the planning of the Gulf of Mexico experiment and, he and Bill Keller showed me how to use microwave radar. David Arnold, the almighty computer programmer, became my loyal companion while conducting the Gulf of Mexico experiment. Francis Felizardo showed me how to use data acquisition boards. He also steered me through the puzzle of courses at MIT. Thanks to Alexei Fedorov for his assistance in the San-Diego experiment. Anatol Rozenberg, the group's "secret" soviet scientist, was instrumental in helping me keep a straight head and a good morale during the Seattle and Woods-Hole experiments, especially when things got tough. His perseverance is admirable.

Special thanks to Mark Loewen who has provided me with continuous support throughout my graduate studies. The list of issues that he has helped me with is too long to mention. I keep fond memories of the Friday afternoons where we would attempt experiments that we had come up with during that week. Our joint experiment at the Texas A&M wave facility remains a memorable success. His practical insight has influenced many aspects of this thesis.

Many undergraduate students have also contributed to my experiments. Fang Chi Wang helped me with modifying the wire wave gauges and showed me how to always pay too much for lunch. Diane Ronan built many electronics circuits. Her good sense of humor always made her fun to be around. Wes Huang did a professional job in designing and building an automated stepper motor system for the wave channel; his work saved me many sleepless nights. Tolga Uzuner made significant improvements to the laser slope meter; I can only wish he would have stayed with us longer. Thanks to Paul Greenberg who carefully analyzed the wave slope data and the void-fraction data acquired during the SWADE experiment offshore Delaware.

The construction of most of the hardware used in the experiments was the work of Jack Crocker and Mario (Mike) Aloisi and his group. I often told Mike that he was charging me too much but I have to admit, his machining work was always first class and on time.

I have enjoyed the Friday afternoons at the Muddy Charles with the "Melville group": Andy, Mark, Gunnar, Francis, Anne, Ge and Analia. Thanks to all of you! To Analia, good luck with your continuing studies *et bon courage.*

My parents deserve the greatest credit. Their continuous and generous support throughout all my studies made it all possible. *A vous deux, Merci.*

I leave graduate life with a fantastic wife which I have been fortunate enough to meet here, in Cambridge. For that alone I shall always be indebted to MIT. Interestingly enough, the one I love is also from Québec and, like two migratory birds, we had to come down south to Massachusetts to reach our common meeting ground. The last three years with her have been most delightful and I can only wish that such blessedness continues.

This research was funded by the Office of Naval Research (Ocean Acoustics and Oceanography) and the National Science Foundation.

Table of content

List of Figures

Figure 2.6: Surface effect. When the free surface is too close to the measuring volume, the void-fraction measured by the instrument is biased by the free surface. The diagram in the upper-right hand corner shows how the depth was measured for the various electrode configurations.

Figure 2.7: Sketch of the field probe used during the SWADE experiment.

Figure 2.8: Calibrations of the field void-fraction probes. The normalized output is given by $V^* = (V_w-V)/(V_w-V_a)$ where V is the voltage output from the instrument, V_w is the voltage in water only and V_a is the voltage in air only. Direct calibrations up to 30% void-fraction were obtained along with calibration at 100% in air. The three different symbols represent calibrations of three different probes. Data is fitted with a second-order polynomial.

Figure 3.1: Sketch of the experimental facility and the instrumentation setup.

Figure 3.2: Wave gauge time-series at a) 3.0m, b) 7.7m and d) 15.5m from the paddle. The smaller plots on the right are the corresponding spectra. Breaking occurs at 8.05m. Time is referenced to initial motion of the wavemaker. Breaking occurs at t=14.4s. This data is for the wave packet S=0.54.

Figure 3.3: Photographs of a breaking wave. The frames progress in time from top to bottom and from left to right. Frame 1 to 9 are separated by 0.2s and frame 9 to 16 are separated by 0.3s. The wave moves from left to right. Each photograph corresponds to a different realization of the breaking wave.

Figure 3.4: Ten wave gauge time-series 5cm upstream of breaking for 10 realizations of the same breaking wave experiment. The larger differences at later times are due to short waves, generated by the breaking wave, which are moving upstream towards the wave gauge. Time is referenced to the initial motion of the wavemaker. Breaking occurs at t=14.4s. This data is for the wave packet S=0.54.

Figure 3.5: Typical time series at 200Hz of a) wave gauge measurements b),c) void-fraction measurements located at depth -5cm and -15cm from the still water level. All data are for the most energetic breaker studied at 0.55m downstream of x_b. Note how the probe crosses the water surface at 0.8sec in b). 'Surface effect' is shown in b) at 3sec. The second wave trough has its water surface at approximately -2.5cm and therefore intrudes into the measuring volume of the void-fraction gauge which is located at -5cm ±2.5cm.

Figure 3.6: Typical repeatability of the void-fraction measurements. The mean of 20 ensemble averages of three repeats (bold line) is plotted along with ± one standard deviation of the 20 ensembles.

Figure 3.7: Color contour maps of the void-fraction field at four different times along with matching photographs of the breaking wave. The center line of the photographs is at 10.5m from the wavemaker paddle. The still water level is the horizontal line. Time is from the start of the paddle motion.

Figure 3.8: Contour maps of the void-fraction field for the most energetic breaker studied. Time evolves from a) through d). The wave moves from left to right. Each frame is constructed from approximately 170 ensemble averages of three runs. Note the distinct downstream cloud generated by the splash from the initial impact at breaking.

Figure 3.9: Moments of the void-fraction field for the 2D experiments. a) Normalized volume of air entrained $V^* = V/V_o$, b) normalized bubble plume cross-sectional area $A^* = A/V_o$, c) mean void-fraction $\bar{\alpha}$, d) potential energy E_b of the plume per unit width, normalized by E_d, the energy dissipated by breaking per unit width. Solid lines are best exponential fits for V^* and E_b/E_d, and best power law fit for $\bar{\alpha}$. The data for A^* has been fitted by using the functional form for V^* divided by that for $\bar{\alpha}$. The values for V_o and E_d used in normalizing V, A and E_b are given in table 3.1. Symbols are S=0.54 (O), S=0.45 (□) and S=0.38 (Δ) where S is defined in table 3.1.

Figure 3.10: Centroids of the bubble plume. a) Horizontal centroid $\bar{x}$ of the bubble plume normalized by λ=1.94m, the characteristic wavelength of the wave packet. b) Vertical centroid $\bar{z}$ normalized by H the height of the wave at breaking (see table 3.1). Solid line is the phase speed of the wave $C = \lambda/T$. Symbols are S=0.54 (O), S=0.45 (□) and S=0.38 (Δ) where S is defined in table 3.1.

Figure 3.11: (On the following page). A series of photographs showing the breakup of the air cylinder. The camera is located inside the wave channel and looking toward the wave paddle for frames 23 to 33 and the breaking wave is coming direclty towards the camera. The time down to milliseconds is indicated on each frame. Frame 34 to 36 are from the side of the channel. In frame 23, the cylinder of air is about to be formed. The lowest horizontal line is the interface between air and water. This particular frame corresponds to frame 35 from the side of the channel. In frame 25, the forward face of the wave impacts the water surface and bubbles start to appear at the impact zone. This corresponds to frame 36 from the side of the channel. In frame 26, we observe a smooth cylinder of air with bubbles at the surface. In frame 27, the cylinder has penetrated the surface even more and instabilities start to appear. In frame 28 and 29, breakup of the cylinder is initiated and small finger-like structures with spacing comparable to the instabilities of frame 27 are formed. Frame 31 and up show the breakup evolving into a complex turbulent bubbly mixture. Video by Eric Lamarre, photos of the video by Mark Loewen [1991]

Figure 3.12: Total volume of air entrained V_o per unit width versus fractional energy dissipated by breaking $D_b = E_d/E_w$ where E_d is the energy dissipated by breaking (without visous loss on the bottom and walls of channel) and E_w is the total energy in the wave packet prior to breaking. The solid symbols are from the present study. The hollow symbols are from three different wave packets studied by Loewen and Melville [1993]. (Adapted from Loewen and Melville [1993]).

Figure 4.1: Top and side view of the multi-directional wave basin facility at the Offshore Technology Research Center (OTRC). The radial lines sketched in the top view represent the wave crests.

Figure 4.2: Front and side views of a section of the array of void-fraction probes used to measure the volume fraction of air. Each probe is made up of three Nichrome wire electrodes 0.127mm in diameter. The wire separation is 1.6cm. The measuring volume for each probe is roughly a cylinder ~5cm in diameter and 20cm in length centered around the midle electrode. The separation between the probes is 15cm. Details of the characteristics of the probe and its calibration are given in §2.

Figure 4.3: Wave gauge measurement for three repeats of the experiment at a) 6.6m upstream of breaking, b) 1.0m downstream and c) 8.6m downstream. At each position, the three plotted repeats are essentially indiscernible and the wave profile shows excellent repeatability. d,e,f) Time-series of wave gauge measurements for two wave gauges placed symmetrically at 4.9m on either side of the basin's centerline at d) 6.6m upstream of breaking, e) 1.0m downstream and f) 8.6m downstream. The figure clearly shows that the wave profile was symmetric with respect to the centerline. The time $t-t_b$ corresponds to the beginning of breaking and T is the characteristic period of the wave packet. All time-series are for a gain G=0.7.

Figure 4.4: Wave gauge time-series measured at six different locations separated by 3m along the basin centerline. Breaking occurs at frame c). All time-series are for G=0.7.

Figure 4.5: Spectra of the time-series shown in figure 4.4a (solid line) and figure 4.4f (dashed line).

Figure 4.6: a) Breaking wave at OTRC (Photograph by R.P. Johnson of OTRC). b) Breaking wave photographed in deep water during the Surface Wave Dynamics Experiment off the coast of Delaware. Both waves begin to break at the center of the wave crest and progressively break from the center towards the edges.

Figure 4.7: Evolution of the whitecap coverage for one of the breaking waves at OTRC (G=0.7). The frames are separated by 0.27s (or 0.135T). The figure was digitized from video images obtained with a camera looking from above. In d), the width of the secondary bubble plume (upper area in d) is greater than the width of the primary bubble plume (lower area in d).

Figure 4.8: Typical time-series at 200Hz of a) wave gauge measurement; b-g) void-fraction measurements at depths 0, -15, -30, -45, -60 and -75cm respectively. Notice the different void-fraction scales for the various plots. Note how the probes cross the water surface in b) and c) and generate large "step like" features in the void-fraction time-series. The probe in c) does not quite cross the water surface during the first wave trough. All time-series are for G=0.7.

Figure 4.9: a),b),c),d) Color contour maps of the void-fraction field. The wave moves from right to left and the corresponding times $t-t_b$ are a) 0.15T, b) 0.25T, c) 0.35T, d) 0.8T. The void-fraction color scale increases by uniform increments of 6% starting at 0.3%. e),f),g),h) Corresponding color contour maps of the sound-speed field. The sound-speed color scale increases by increments of 20m/s and corresponds to the range covered by the void-fraction scale. The breaking wave shown is for G=0.7.

Figure 5.7: (Prevous page) Photographs of the bubble plume taken from the video recording. The photographs evolve in time from left to right and from top to bottom. Frame 1 to 7 are 0.133s apart, frame 7 to 12 are 0.267s apart and frame 12 to 15 are 0.533s apart. Frame 8 corresponds to the arrow on figure 5.3d. The largest bubble (5mm radius) in this sequence is seen in frame 7.

Figure 6.1: Sketch of the buoy and its instrumentation.

Figure 6.2: Sketch of the support module.

Figure 6.3: Bloc diagram of transmit acoustic system.

Figure 6.4: Bloc diagram of receive acoustic system.

Figure 6.5: a) time-series of the pulse transmitted to the power amplifier. b) Spectrum of time-series in a). The time axis in a) and frequency axis in b) are normalized by the period T and the frequency f_o of the acoustic wave.

Figure 6.6: a) time-series of transmit (above) and receive (below) 5kHz acoustic pulses. b) Spectra of time-series in a). Note that the receive time-series was windowed with a 500 point Bartlett window in order to eliminate the reflections from contributing to the spectrum. The second line on the lower spectra is the ambient sound for $W_s = 8$m/s.

Figure 6.7: a) time-series of transmit (above) and receive (below) 10kHz acoustic pulses. b) Spectrum of time-series in a). Note that the receive time-series was windowed with a 250 point Bartlett window in order to eliminate the reflections from contributing to the spectrum. The second line on the lower spectra is the ambient sound for $W_s = 8$m/s.

Figure 6.8: a) Cross-correlation function for the time-series in figure 6.6a (i.e. 5kHz pulses). b) Expanded section around the peak of a). c) Expanded section around the peak of b). The data from the cross-correlation are shown as (o) in b) and c). The solid line in b) and c) is the interpolation.

Figure 6.9: a) Cross-correlation function for the time-series in figure 6.7a (i.e. 10kHz pulses). b) Expanded section around the peak of a). c) Expanded section around the peak of b). The data from the cross-correlation are shown as (o) in b) and c). The solid line in b) and c) is the interpolation.

Figure 6.10: a) Sketch of the laboratory tank and of the position of the hydrophones for the interference test. b) Sketch defining the position of the rod for the interference test.

Figure 6.11: Pulse distortion from interference by a solid rod placed parallel and close to the direct propagation path. For each frame, the top plot is the transmit pulse and the bottom plot is the received pulse. a) 10kHz, b) 25kHz and c) 50kHz. At each frequency, three configurations were tested. 1) No rod, 2) d=2cm and 3) d=8cm. The separation d between the rod and the head of the hydrophones is indicated on the plots when the interference is noticeable.

Figure 6.12: Test of acoustic interference by the support module at 5kHz. The experiments were conducted in a tank 3m in diameter and approximately 10m deep. a) The hydrophones were supported as in figure 6.10. b) The hydrophones were mounted on the support module of figure 6.2. The transmit and receive pulses are shown above and below respectively.

Figure 6.13: Test of acoustic interference by the support module at 20kHz. The experiments were conducted in a tank 3m in diameter and approximately 10m deep. a) The hydrophones were supported as in figure 6.10. b) The hydrophones were mounted on the support module of figure 6.2. The transmit and receive pulses are shown above and below respectively.

Figure 6.14: Pulse distortion caused by bubbles on the transducers. Transmit pulses (above) and receive pulses (below). The acoustic frequency is 10kHz.

Figure 6.15: Cramer-Rao lower bound on the standard deviation of the time-delay estimate σ_D for an observation time T=0.0005s, a signal frequency $f_c=(f_1+f_2)/2=5$kHz and a signal bandwidth $f_2-f_1=3$kHz. Note that results are also shown for 1kHz and 6kHz bandwidths. For the laboratory, SNR~1000 and in the field SNR~100.

Figure 6.16: Modified support for varying the distance between the transmit and receive hydrophone.

Figure 6.17: Changes in transmitter-receiver separation. The separation distance was varied between 400mm and 425mm. Only the change in separation Δs with repect to the 400mm reference is plotted. a) 20kHz, b) 5kHz. Δs_a measured with a micrometer. Δs_{td} measured from changes in time-delays. Actual measurements (o) and residual (•). Note that the vertical scale for the residual is on the right hand side of the graphs.

Figure 7.7: a) Correlation coefficient between measurements at 0.5m and 0.75m (ρ_{1-2}); and between measurements at 0.5m and 1.0m (ρ_{1-3}). b) Coherence between measurement at 0.5m and 0.75m (γ^2_{1-2}); and between measurements at 0.5m and 1.0m (γ^2_{1-3}). The average of three 20min time-series were used for both plots. The time-series were digitally low-pass filtered at 0.1Hz for the results shown in a) in order to eliminate the high-frequency wave-generated signal which is in-phase at all depths and which distorted the peak near zero lag. Wind speed and significant wave height were 8m/s and 0.45m respectively. Woods Hole data.

Figure 7.8: a) Probability distribution of the sound-speed anomaly at 0.5m for three 20min record (f=10kHz). The bin size used for the distribution is 5m/s. b) Same as in a) but plotted with three other depths on a log-linear scale. The straight lines show the trend of the data away from the noise region. The distribution at 2.0m is representative of the measurement's noise. c) Cumulative probability distribution corresponding to the distributions in b). The dotted line in c) shows that at a depth of 0.5m, 20% of the data had a sound-speed anomaly greater than 27m/s. Woods Hole data.

Figure 7.9: Average sound-speed anomaly $\overline{\Delta c}$ as a function of depth. Each data symbol is a 20min average. (∇) Woods Hole experiment and (o) San Diego experiment. The solid line is the best exponential fit to the data. Wind speed of 7 to 8m/s. The frequency of the acoustic pulses was either 5kHz or 10kHz.

Figure 7.10: a) Probability distribution for three different wind conditions. Each distribution was obtained from three 20min time-series. (o) 7.5 to 8m/s (WH), (∇) 5.5 to 6m/s (SD), (□) less than 3m/s (SD) (representative of the noise level). b) Cumulative probability distribution for the same data.

Figure 7.11: Average sound-speed anomaly at 0.5m depth as a function of a) wind speed and b) wind speed normalized by the phase velocity of the dominant waves C_{wave}. Data were averaged over 20min records. Solid line in a) is best linear fit to the data with a 0.74 correlation coefficient. (o) Data from San Diego experiment, (∇) data from Woods Hole experiment.

Figure 7.12: Average sound-speed anomaly at a depth of 0.5m as a function of the significant wave height for the Woods Hole data. Data were averaged over 20min records.

Figure 7.13: Highest sound-speed anomalies measured during both experiments at a)0.5m and b)0.75m. The first large peak at approximately t=50s was caused by a wave which broke at the buoy at the time indicated by the arrow. The second peak approximately 25s later was caused by a bubble cloud formed 25s earlier which drifted by the buoy. The void-fraction corresponding to an 800m/s anomaly is shown on the left axis. The frequency of the acoustic pulses was 5kHz. Woods Hole data.

Figure 7.14: a) Transmit (above) and received (below) composite pulses. The filtered version of the composite pulse shown in a) with band-pass filter center frequencies b)6kHz, c)10kHz, d)20kHz and e)40kHz. The window function plotted with each transmit pulses was used to window the signal and thus eliminate the tail-end of the transmit pulses from biasing the cross-correlation function. Woods Hole data.

Figure 7.15: Spectrum of a) the transmit composite pulse and b) the received composite pulse shown in figure 7.14a. Note the specific peaks in a) at 5, 10, 20 and 40kHz. Same peaks are found in b) except at 5kHz where the transmit hydrophone did not put out enough power.

Figure 7.16: Schematic of the measurement of the pulse amplitude. Transmit pulse above and received pulse below. First, the time-delay τ is computed from the cross-correlation. The closest peak located at a time-interval τ away from the dominant peak in the transmit signal is located next (labeled 1). Then, the closest trough ahead of the peak labeled 1 is finally located (labeled 2). The amplitude between the trough and the peak characterizes the amplitude of the pulse.

Figure 7.17: Sound-speed anomalies at various frequencies obtained by band-pass filtering a 20min record of composite pulse data acquired at a depth of 0.5m. The various band-pass frequencies are indicated on the plots. The axes on the right-hand side show the corresponding void-fraction. The wind speed and the SWH were 8m/s and 0.46m respectively. Woods Hole data.

Figure 7.18: Attenuation at various frequencies obtained by band-pass filtering a 20min record of composite pulse data acquired at a depth of 0.5m. The various band-pass frequencies are indicated on the plots. The wind speed and the SWH were 8m/s and 0.46m respectively. Woods Hole data.

Figure 7.19: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency. The averages were computed over the 20min time-series shown in figure 7.17 and figure 7.18. Note how $\overline{\Delta c}$ is almost independent of frequency below 20kHz.

Figure 7.20: Standard deviation of a) the sound-speed anomaly $\sigma_{\Delta c}$ and b) attenuation σ_A as a function of frequency. The standard deviations were computed over the 20min time-series shown in figure 7.17 and figure 7.18.

Figure 7.21: Sound-speed anomaly for 10kHz acoustic pulses at a depth of a)0.5m, b)0.75m and c)1.0. Attenuation for 40kHz acoustic pulses at a depth of d)0.5m, e)0.75m and f)1.0m. Same data record as the one used in figure 7.17 and figure 7.18.

Figure 7.22: Cumulative probability distributions of the sound-speed anomaly at a depth of 0.5m for a) three low frequencies and b) three different frequencies spanning the range of the measurements. The dotted line shows that 30% of the time the signal was above 24m/s. Same data record as the one used in figure 7.17 and figure 7.18.

Figure 7.23: Cumulative probability distributions of the attenuation at a depth of 0.5m for a) three high frequencies and b) three different frequencies spanning the range of the measurements. The dotted line shows that 30% of the time the signal was attenuated by more than 12dB/m. Same data record as the one used in figure 7.17 and figure 7.18.

Figure 7.24: A conceptual model of the bubble population showing various slopes and critical radii which define the shape of the population. The number of bubbles per m^3 per μm increment is given by n(a) and N is that number at the peak. The subscripts 'p', 't' and 'c' stand for peak, transition and cutoff

Figure 7.25: Sensitivity to variations in bubble population parameters for a) r_c, b) s_4, c) s_3 and d) r_t. The standard deviation σ between the model and the measurements is given by equation 7.6. In each plot, the default bubble population used is given in table 7.4, and one of its parameters is varied.

Figure 7.26: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency (from figure 7.19). The solid line in both plots is obtained by integrating equation B.18 and B.19 with a bubble population described by the parameters in table 7.4 and shown in c). Note that the horizontal axes in c) show both the bubble radius and the corresponding bubble resonance frequency. The number density of bubbles per unit micron increment is n(a). The solution was optimized to minimize the variance between the model and the data according to equation 7.6.

Figure 7.27: Sound-speed anomaly $\overline{\Delta c}$ (above) and attenuation $\overline{A}$ (below) as a function of frequency for different variations in the bubble population parameter a) r_c, b) s_4, c) s_3 and d) r_t. The starting bubble population is the one given in table 7.4 and figure 7.19c. Only one of the bubble population parameter is varied in each of the various graphs above. The symbol (o) corresponds to the original value and the symbols (∇) and (□) are the variations above and below the original value.

Figure 7.28: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency (from figure 7.19). The solid line in both plots is obtained by integrating equation B.18 and B.19 with a bubble population shown in c). Note that the horizontal axes in c) show both the bubble radius and the corresponding bubble resonance frequency. The number density of bubbles per unit micron increment is n(a). The solution was optimized to best fit only the sound-speed anomaly. The bubble population parameters are $s_1=2$, $s_2=-4$, $s_3=-2.5$, $s_4=-6$, $r_p=20\mu m$, $r_t=50\mu m$ and $r_c=140\mu m$.

Figure A.1: Low frequency sound-speed as a function of void-fraction. a) linear-linear scale. b) log-linear scale. The lower curve on both plots is for atmospheric pressure and ambient temperature. The upper curve is for atmospheric pressure and a hydrostatic pressure corresponding to a depth of 2m and ambient temperature.

Figure A.2: Low frequency sound-speed as a function of void-fraction at very low void-fractions. The solid line is from the exact equation A.6 and the dashed line is from the linear approximation given by equation A.7. Note that the graph has linear-linear scales.

Figure B.1: Dispersion of sound phase velocity for a uniform bubble population. The frequency f is normalized by the resonance frequency of the bubbles f_R. The sound-speed anomaly $\Delta c=c-c_w$ is normalized by the sound-speed in bubble free water c_w. The data shown is for bubbles of size $100\mu m$ and a number of bubbles per m^3 N=100 and N=1000. The low-frequency asymptote is a function of the void-fraction and the high-frequency asymptote is the speed of sound in bubble-free water.

Figure B.2: Plot of the kernels of equations B.18 and B.19 (or B.17). The upper, middle and lower curves on each plot are for bubbles of radius $3250\mu m$, $325\mu m$ and $32.5\mu m$ (f_R=1kHz, 10kHz and 100kHz resonance frequencies) respectively. At frequencies much below bubble resonance $K_{re}=1$ while much above bubble resonance $K_{re}=0$. K_{im} is zero both above and below bubble resonance. Dispersion and attenuation effects are generated at or near bubble resonance.

Figure C.1: Computed breaking wave packet used in the 2D experiment of §3. High frequency waves are generated first followed by lower frequency waves. Dispersion ensures superposition at a predetermined position x_b from the wave paddle. A PC equipped with a Metrabyte Das20 board performed the D/A at a conversion rate of 100Hz. The analog signal was fed to the input of the wavemaker system. Note that the packet shown has been corrected for the wavemaker transfer function.

Figure C.2: a) Amplitude and b) phase transfer function for the MIT wavemaker system installed on the wave channel 25m $\times$ 0.7m $\times$0.6m. The hollow symbols are for an input amplitude of 0.1V and the filled symbols are for an input amplitude of 0.2V. Transfer function measured on 08-24-90.

Figure C.3: Wave channel settling time. The standard deviation of wave gauge displacement measurements computed over 20s windows is shown as a function of the time after onset of breaking. The channel reaches a steady calm state approximately 6min after onset of breaking. The 5 different symbols represent 5 different repetitions of the breaking event.

Figure C.4: The fractional dissipation of the wave packet energy as a function of the slope parameter S. The data shown is for the wave packet with center frequency f_c=0.88Hz, bandwidth Δf=0.88Hz and normalized breaking location $x_b k_c$=24.6 where k_c is the wavenumber of the center component of frequency f_c.

Figure D.1: Model used in the study of the effects of sampling rate and interploation (discussed in appendix E). The peak of the cross-correlation function is modelled as a half cosine. The symbols represent the actual samples which define the cross-correlation peak. The solid symbols are the data selected for the spline fit (i.e. for interpolation)

Figure D.2: Error $\Delta\tau_{err}$ on the resolution of the location of the peak of the cross-correlation function normalized by the acoustic wave period T. The sampling rate is expressed as the number of data samples per period. a) Amplitude noise A_m and b) phase noise ϕ_n. A 5 point spline interpolation was used in all simulations.

Figure E.1: Error $\Delta\tau_{err}/T$ in the resolution of the location of the peak of the cross-correlation function versus the interpolation factor. T is the acoustic wave period. The number of samples per period and the number of points used in the spline are shown. The sampling rate is expressed as the number of data samples per period.

Figure G.3: Circuitry for converting the frequency modulated sine wave output (f) of the SBE-3 temperature probe to a TTL compatible signal of frequency f/256. The TTL signal is used to drive a Metrabyte CTM-PER period counter board which extracts the frequency modulated information contained in the TTL signal. Circuit adapted from Horowitz and Hill [1989, pp242].

Figure G.4: a) Water admittance of ITC-1042. b) Transmitting Voltage Response of ITC-1042.

Figure H.1: Bloc diagram of programs organization for processing and plotting the bubble plume data of the 2D experiment (§3). Note that the processing of the data for the 3D experiment (§4) is essentially similar. Bold boxes are programs, all other boxes are input files, output files or subroutines. The (*) refers to various file numbers.

Figure I.1: Bloc diagram of programs organization for the data acquisition system used in §6 and §7 for the measurements of sound-speed. Bold boxes are programs, all other boxes are input files, output files or subroutines. The (*) refers to various file numbers.

List of Tables

Table 5.1: Three deployments during the SWADE field test. Wind speed, significant wave height (SWH) and dominant wave period (T_{dom}) are from nearby SWADE buoys located no further than 8km away. Wind speed was measured at 4m and estimated at 10m (U_{10}) using a logarithmic wind profile with a drag coefficient for neutral conditions [Large and Pond, 1981]. Wind speed was averaged over 8min, significant wave height and dominant wave period are 20min averages. The fraction of breaking waves per wave (β) is obtained from the number of void-fraction (V.F.) events above 1% divided by the duration multiplied by the dominant wave period. Eastern Standard Time (EST) is with respect to the center of the hourly record.

Table 6.1: Maximum possible length of the direct path as a function of acoustic frequency and hydrophones depth. The transmit and receive hydrophones are assumed to be at the same depth. The length of the acoustic pulse $L_p=2\lambda$ where λ is the acoustic wavelength.

Table 6.2: Signal generator and power amplifier gain settings for the two field experiments described in §7. The voltage gain of the signal generator is applied to the transmit signal of maximum amplitude of unity (i.e. a gain of 5 means that the signal generator will output a pulse of 5V peak amplitude.) The composite pulse was composed of 4 distinct frequencies. It is described in detail in §7.

Table 6.3: Gain of pre-amplifiers and amplifiers, and cutoff frequencies of the high-pass and low-pass filters for various acoustic frequencies. The settings for the two field experiments described in §7 are shown. The composite pulse was composed of 4 distinct frequencies. It is described in detail in §7.

Table 6.4: The three principle bloc components of the data acquisition system.

Table 7.1: Characteristics of the digital FIR band-pass filters used to filter the composite pulses. The 3dB points of the band-pass filter are located at f_{low} and f_{high} and the center frequency is f_c. The number of points in the filter is N.

Table 7.2: Range of parameters surveyed in the search of the best fit solution to the measurements of the attenuation and propagation speed.

Table 7.3: Values of k_{re} and k_{im} for the data shown in figure 7.19. Note how $k_{im} << k_{re}$. The values for $|\Delta c|$ are given by equation 7.5.

Table 7.4: Optimal bubble population parameters obtained from the computational survey of all parameter combinations given in table 7.2.

Chapter 1
Introduction

Breaking waves charge the surface layer of the oceans with small bubbles which play a significant role in many upper-ocean processes. The present study focuses on describing and quantifying the entrainment of air by breaking waves and its effect on the dynamics of the upper-ocean.

The introduction begins with an overview of the role and importance played by the mixed layer in general, and by the surface layer in particular, in regulating exchanges between the atmosphere and the ocean. We then move on to give literature reviews of specific research areas related to this thesis which should place the contribution of the present work in the appropriate context.

1.1 The mixed layer and the ocean surface layer

A relatively shallow layer of water, referred to as mixed layer, isolates the ocean's interior from the atmosphere. The depth of this mixed layer depends on surface conditions[1] and can range from a few to several hundreds of meters. In general, this layer is well mixed by wind and waves which generate the necessary turbulence to stir the water. Hence, temperature, salinity and other properties are nearly uniform within the mixed layer.

On a depth scale an order of magnitude smaller, breaking of surface waves can entrain air bubbles below the water surface and form a bubbly layer or what is also known as the surface layer. The depth of the surface layer is also a function of surface activity (i.e. wind and waves) and usually it will be on the order of meters.

The surface layer plays a crucial role in the exchange of **energy**, **mass** and **momentum** between the atmosphere and the ocean's interior. For example, 40% of the solar radiation incident on the atmosphere is absorbed within the first 10m of the ocean and the upper 2.5m of the ocean can store as much heat as the entire atmosphere [Gill, 1982]. Therefore, by absorbing, transporting and returning heat to the atmosphere, the upper ocean greatly affect the weather and the climate [Weller and Farmer, 1992].

[1] Surface conditions include, among other things, wind speed, wave height, air and water temperatures.

The oceans are the second largest reservoir of carbon after the earth's crust. The largest fluxes of CO_2 between any two reservoirs of carbon occur between the ocean and the atmosphere [Sundquist, 1993]. These fluxes must necessarily transit through the surface layer and the rates at which they do is strongly influenced by the dynamics of that layer. Hence, our ability to predict the long term effects of anthropogenic (produced by human) CO_2 on our environment is going to be, to a large extent, a function of our ability to accurately model the exchanges through the surface layer. Other important mass transfers through the surface layer include precipitation and evaporation of fresh water which accounts for an important part of the global hydrological cycle and which play a vital role in the atmospheric heat balance and in the ocean thermohaline circulation [Gill, 1982].

Momentum is also transferred across the air-sea interface. Wind blowing over the ocean generates waves and shear currents which in turn stir the entire mixed layer. This momentum transfer is an essential part of the ocean circulation along with thermohaline forcing which is caused by heat exchange. Thus, the surface layer plays a critical role in transferring the momentum from the wind and waves at the surface to the water column below.

It is clear from the few examples given above that the surface layer plays an important part in regulating the transfers of energy, mass and momentum between the atmosphere and the ocean's interior. Hence, it is no surprise that much recent work has been and is conducted in the field of air-sea interaction.

1.2 Air bubbles within the surface layer

One of the most notable features of the surface layer is the presence of air bubbles which are entrained in the water column by the breaking of surface waves[2] . Although some bubbles have been observed down to a depth of ~10m, usually they are located within one or two meters of the water surface [Thorpe, 1982; Farmer and Vagle, 1989; Zedel and Farmer, 1991]. The bubbles spatial distribution is highly variable and in general the bubble density decreases with depth and increases with sea state [Crawford and Farmer, 1987]. Furthermore, recent evidence suggests that the bubbles, at least the very small ones, are entrained into vertical pairs of counter-rotating eddies known as Langmuir cells [Thorpe, 1982; Zedel and Farmer, 1991; Smith, 1992; Weller and

[2] In general, the onset of wave breaking in the open ocean occurs at wind speeds of 3 to 4m/s.

Farmer, 1992; also refer to Thorpe, 1992 for a review of bubble clouds and Langmuir circulation measurements by sonar]. Langmuir cells have their axis of rotation approximately aligned with the wind and small bubbles get carried to depth at the convergence zone of the counter-rotating vortices.

Air bubbles have an important role in the dynamics of the surface layer because of their ability to increase the effective transfer rate of heat and gases between the atmosphere and the ocean's interior (§1.6). Bubbles can also scavenge organic and inorganic particles [Blanchard and Syzdek, 1972; Wallace and Duce, 1978] which can be transported and ejected back to the atmosphere as aerosols (spray) by the bursting of the bubbles at the surface [Blanchard and Woodcock, 1957, Blanchard, 1983]. Air bubbles also degrade the performance of ships' sonar when operated close the sea surface [Dalen and Lovick, 1981].

It is also well known that bubbles scatter and absorb sound as well as being a significant source of sound at their formation (§1.7). In the ocean, bubbles have been found to form distinct clouds which are the result of breaking waves and Langmuir circulation [Thorpe, 1982; Farmer and Vagle, 1989]. These bubble clouds are believed to be responsible for the high levels of acoustic reverberation observed at the ocean surface at wind speed in excess of 10m/s [McCammon and McDaniel, 1990; Henyey, 1991; McDonald, 1991; Rino and Ngo, 1991].

Acoustic waves propagate at a velocity of 1500m/s in bubble-free water while in air they propagate at 340m/s. In the presence of bubbles, the velocity of sound in water can be significantly altered especially if the frequency of the acoustic wave is close to the natural resonance frequency of the bubbles (refer to appendix B). At frequencies below bubble resonance (say below 20kHz), the sound velocity is no longer dominated by bubbles excited at their natural resonance frequencies. Instead, it is the total volume fraction of air in the water (hereafter void-fraction) that controls the propagation speed of the acoustic waves. In fact, void-fraction and sound-speed at low frequencies are intimately related through Wood's equation [Wood, 1941] (refer to appendix A for details) and indeed you can always compute one from the knowledge of the other. For example, void-fractions of 10^{-6}, 10^{-4} and 10^{-2} correspond to sound-speed reductions of 100m/s, 800m/s and 1400m/s respectively. These sound-speed reductions are very large compared to fluctuations caused by changes in temperature, salinity or density which typically range from a few to tens of meters per seconds.

Sound is a powerful tool which can be used to remotely monitor the oceans. In fact, underwater sound is the only tool presently available to probe the oceans at range greater than a few meters and the science of using acoustic technique to study oceanographic processes has developed into a distinct and rapidly evolving field called *acoustical oceanography* [Clay and Medwin, 1977]. Some example of the use of acoustics in the ocean include: underwater communications; mapping of the ocean floor; detection of underwater vehicles or animals; geophysical exploration and more recently, ocean tomography which may one day give us information about ocean warming trends. All these examples fall in the category of active remote sensing. Passive techniques include the monitoring of wind speeds [Vagle et al., 1990] and rainfall rates [Nystuen, 1986; Laville et al., 1991] over the oceans and the detection of breaking waves [Ding, 1992]. It is also hoped that in a near future we may be able to remotely sense the dissipation of surface wave energy by breaking [Loewen and Melville, 1991a] and the transfer rates of gases across the interface (§3).

It is the distinct underwater acoustic signature of breaking waves that makes passive remote sensing of upper-ocean processes possible except for the case of rainfall rates where clearly the mechanism is the rain itself and not wave breaking. Indeed, the wind-dependent natural ambient sound in the ocean is known to be dominated by breaking waves. This issue will be discussed further in §1.7.

Clearly, sound is an invaluable tool for learning about and monitoring the oceans and the processes occurring at their surface. In order to fully exploit this capability, especially near the ocean surface, much is to be gained by understanding how sound propagates in the surface layer; the speed at which it travels; its attenuation and scattering characteristics. Since the rough water surface and the air bubbles below it dominate the nature of sound propagation near the surface; it is imperative to have a good description of the spatial and temporal changes in the sound-speed field (and hence in the void-fraction field) within that bubbly region, and to relate these changes to the wind and wave forcing at the surface. These issues are discussed in more detail in §1.8.

Air bubbles entrained by breaking waves therefore play an important role in the transport of mass and energy across the air-sea interface (although the latter is yet to be shown conclusively). Furthermore, bubbles play a central role in the underwater acoustics of the upper-ocean.

1.3 Breaking waves

Wave breaking plays an important role in a number of processes at the air-sea interface. These include [Melville, 1992]:

- limiting the height or surface waves by dissipating wave energy [Rapp and Melville, 1990],
- being a source of turbulence and enhancing mixing [Agrawal et al., 1992],
- generating ocean currents by transferring momentum from the wave field [Melville and Rapp, 1985],
- enhancing energy and mass transfers by bubble entrainment [Merlivat and Memery, 1983] and turbulence [Khoo and Sonin, 1992],
- generating ambient sound which can be used in passive acoustic remote sensing [Loewen and Melville, 1991a],
- providing a distinct microwave signature which can be used in radar remote sensing [Jessup et al., 1991b].

Since breaking is the dominant mechanism by which air is entrained in the surface layer[3], it would seem appropriate to review the current knowledge of the topic. This literature review is therefore tailored to emphasize the areas of research more relevant to the present study. For a broader literature review of (deep water) breaking waves, the reader is referred to Melville [1992] and Banner and Peregrine [1993].

1.3.1 Detection and quantification of breaking waves in the field

The most common expression of breaking is the generation of a foamy whitecap caused by the entrainment of air bubbles [Monahan and McNiocaill, 1986 for a recent proceeding on whitecaps]. However, small scale breaking of very short gravity wind waves also occurs at sea without any air entrainment and associated whitecaps. This lack of a universal visual feature has made advances in the detection and quantification of breaking waves in the field more difficult.

Several experimental techniques have been used to obtain breaking wave statistics. Among them are: a) the visual presence of a whitecap at a fixed point [Holthuijsen and Herbers, 1986]; b) the time-derivative of the surface elevation at a wire wave gauge [Thorpe and Humphries, 1980; Longuet-Higgins and Smith, 1983]; c) the high-frequency spectral content from surface elevation measurements [Weissman et al., 1984;

[3] Note that rain can also entrain air bubbles.

Katsaros and Ataktürk, 1991]; d) the microwave Doppler radar return from the sea surface [Jessup et al., 1991a,b]; e) and the use of ambient sound to detect breaking [Farmer and Vagle, 1988; Farmer & Ding, 1992; Ding, 1992].

Figure 5.5 in the present study shows many of the results obtained in the studies cited above on the fraction of breaking waves per wave. In general, all of the above techniques show distinctive trends with wind speed but significant scatter exists when comparing results obtained with different techniques. While some of the scatter may be due to different ocean conditions (i.e. fetch, water depth and others) it is generally believed that most of the differences across studies is caused by dissimilar detection techniques and their associated definition of breaking [Banner and Peregrine, 1993].

Chapter 5 of this work reexamines these issues and presents results from a novel measurement technique for the detection of breaking waves in the lab and in the field.

1.3.2 Laboratory studies of breaking waves

The temporal and spatial intermittency of breaking waves makes their study in the field extremely difficult. Furthermore, because of the complexities of the two-phase turbulent flow associated with wave breaking, most of our quantitative knowledge of the subject comes from laboratory experiments with controlled breaking waves. Here we review in detail two such studies which have significant bearing on this thesis. They focus on the energy dissipation and acoustic radiation from breaking waves.

Melville and Rapp [1985] used the dispersive properties of deep-water waves to focus a wave packet in a laboratory channel (see appendix C for details on the generation of breaking waves). With this technique, they could vary the strength and position of the breaking wave in the channel and obtain extremely repeatable breaking events. They made surface displacement measurements upstream and downstream of breaking and computed the momentum flux of the wave as it progressed down the channel. The momentum flux is calculated by computing the surface displacement variance; and the dissipation caused by breaking is found by taking the difference in energy flux between upstream and downstream measurements. Figure 1.1 shows measurements of the momentum flux as a function of position along the channel. Clearly, there is an abrupt loss caused by breaking. These laboratory studies were considerably expanded by Rapp and Melville [1990] where they reported that up to 40% of the initial energy contained in the wave field could be dissipated by breaking.

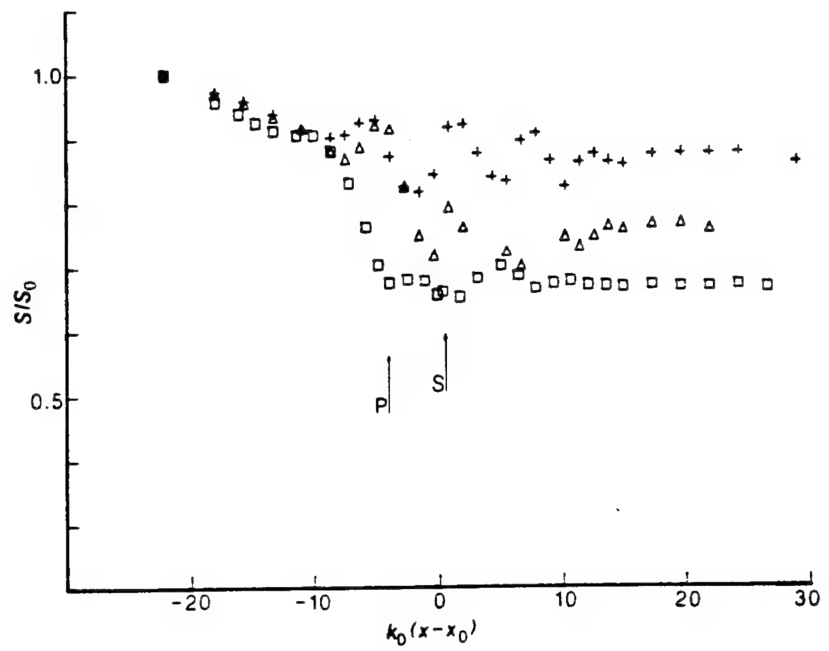


Figure 1.1: Laboratory measurements of the momentum flux S as a function of downstream position. Breaking occurs at $x=x_o$. The position is normalized by the characteristic wavenumber k_o of the breaking wave and S is normalized by the upstream momentum flux S_o. The steady initial decrease in S is caused by viscous dissipation on the walls and bottom of the channel. Note the decrease in S across the breaking region. Also note that $S=E/2$ where $E=(\rho_w g a^2)/2$ is the wave energy density. Symbols refer to three different breaking wave intensities. From Melville and Rapp [1985].

Clearly, the above measurements support the fact that breaking contributes significantly to the transfer of energy from the wind to the upper-ocean. However, what may not be so obvious is the partitioning of the energy transferred. For example, is the transferred energy mostly used to generate current or is it expanded in entraining air bubbles below the surface? Perhaps air entrainment is not very significant and most of the energy is immediately lost to turbulence. These issues are addressed in §3 where it will be shown that air entrainment takes a significant fraction of the energy lost by breaking.

The second series of laboratory measurements to be reviewed are those of Melville et al. [1988] and Loewen and Melville [1991a] on the acoustic radiation from breaking waves. Their measurements were conducted with breaking wave packets similar to those used by Rapp and Melville [1990]. They found that the sound radiated by breaking waves at frequencies greater than 500Hz correlated with the energy dissipated by breaking. Their finding was significant since it raised the prospect that dissipation of ocean breaking waves could be quantified acoustically, a very difficult and yet not feasible measure to make directly.

These results also raised the issue that perhaps other surface processes such as air entrainment could be monitored acoustically. For example, the measurements of Medwin and Daniel [1990] on gently spilling laboratory breaking waves had shown that the individual bubbles entrained by breaking could be measured and counted using their damped sinusoidal acoustic signature. Of course, this technique becomes impracticable when large volumes of air are entrained such as the ones found under energetic breaking waves. Nevertheless, these larger breaking waves have a distinct acoustic signature which may, as with the gentle spillers, be correlated to an integral measure of the volume of air entrained. This issue is addressed in §3.

1.4 Bubble population in the ocean

Although no bubble population measurements are reported in the present work, previous measurements are nevertheless reviewed because of their relevance in inferring void-fraction and low-frequency sound-speed in the ocean.

1.4.1 Measurements of bubble population

In a well-developed sea one can hypothesize that the production rate of bubbles by breaking waves reaches a steady state and likewise, the bubble density sustained by the upper-ocean must reach an equilibrium. The production rate of bubbles at the surface, the downward momentum transferred to them by breaking, their rise velocities and their dissolution time constants are the major mechanisms responsible for establishing and maintaining this dynamic equilibrium.

Authors	range (μm) and technique	slopes	peak (μm)	meas. depth & water depth (m)	depth dependence of total number of bubbles	num. of bubbles at a=100μm per m^3 per μm incr.	wind sp. range (m/s)	wind speed dependence
Medwin [1970]	18 to 180 acoustic	a^{-4} a<80 a^{-2} a>80	no peak	1.5 to 14 coastal waters	$N\sim e^{-z/L}$ a<60μm L=7m $N\sim z^{-1/2}$ a>60μm	150 d=1.5m U=1-4m/s	1 to 4	N/A
Kolovayev [1976]	14 min. bubble trap	$a^{-3.5}$	70 to 80	1.5 to 8 open sea	$N\sim e^{-z}$ z<3m * $N\sim z^{-2.6}$ z>3m *	350 d=1.5m U=11-13m/s	6 to 13	$N\sim U^{4.5}$ *
Medwin [1977]	15 to 300 acoustic	a^{-4} a<60 a^{-2} a>60	no peak	3 to 36 1000	none	250 d=4m U=6m/s	up to 6	N/A
Walsh and Mulhearn [1987]	50 min. photography	a^{-4} a>100	68	0.5 to 2 120	observations showed dependence with depth	300 d=1.0m U=9m/s	2 to 14	$N\sim U^{3.3}$ U<6m/s
Johnson and Cooke [1979]	17 min. photography	a^{-5}	40 to 50	0.7 to 4 20 to 30	$N\sim e^{-z}$ z<3m * $N\sim z^{-2.6}$ z>3m *	176 d=1.8 U=11-13	8 to 13	$U^{4.5}$ *
O'Hern et al. [1988]	10 to 250 holography	a^{-4} 10<a<50	no peak	3 to 33 200	N/A	N/A	1 to 4	N/A
Su [1988]	20 to 400 optical	a^{3} a<30 a^{-5} to a^{-6} for a>200	40 to 50	2 to 15 30	none	1000 d=2m U=9.5-10.5m/s	2 to 18	$N\sim U^{4.0}$ to $U^{4.6}$
Baldy [1988]	30 to 1500 laser laboratory	a^{-4} a<50 $a^{-2.6}$ a>50	no peak	0.05 to 0.25 1	$n(a)\sim a^{-3.3}$ at 0.05m $n(a)\sim a^{-3.9}$ at 0.25m	1000 d=0.25m U=14m/s	11 to 14	$n(a)\sim a^{-4.8}$ 11m/s $n(a)\sim a^{-3.3}$ 14m/s
Medwin and Breitz [1989]	30 to 240 acoustic	a^{-4} a<60 $a^{-2.5}$ a>60	no peak	0.25m 500m	N/A	1000 d=0.25m U=12to15m/s	0.25	N/A
Vagle [1989]	16 to 116 acoustic	a^{-4} to a^{-6} for a<peak	16 to 37 140 or greater	continuous 0.1 to 10m 140m +	N/A	200 d=1m U=11m/s	0 to 16	N/A
Hwang et al. [1990]	800 to 3000 laser laboratory	$n(a)\sim a^{-\alpha}$ α=2.8z/Hs+0.7 0.5<z/Hs<1.2	no peak	6, 7, 9, 10, 15cm 0.75m depth	$n(a)\sim a^{-\alpha}$ α=2.8z/Hs+0.7 0.5<z/Hs<1.2	N/A	lab 10 to 15	N~exp(-z/Hs)
Vagle and Farmer [1992]	8 to 130 acoustic	varriable a^{-4} to a^{-7} for a<25	25	continuous 0.1 to 10m 140m +	N/A	1000 d=0.5m U=11m/s	data for 11m/s	N/A
Su et al. [1993]	34 to 1200 acoustic	a^{-4} a<80 a^{-3} a>80 up to 1200	no peak	6 depths starting 0.24m 90m depth	N/A	2000 d=0.24m U=10m/s	10 to 15m/s	N/A
Kalvoda [1993]	1 to 5mm video laboratory	$a^{-3.8}$	N/A	in plume 0.75m depth	N/A	N/A	lab 16m/s	N/A

Table 1.1: Review of bubble population data. The (*) refers to Wu's review paper [1981]. Peak and slope refer to the shape of the bubble population. The total number of bubbles per m^3 is N and n(a) is the number of bubbles per m^3 in a band da (1μm). N/A refers to "not applicable or available" and "none" means no dependence in the data was found.

Knowledge of the bubble population is necessary for any adequate modeling of the contribution of bubbles to heat and gas transfer. This is, in part, why several studies have been conducted over the last twenty years on quantifying the population of bubbles in the surface layer of a wind-driven ocean and in laboratory wind-wave tanks. The measurement techniques have included acoustic backscatter and attenuation, bubble traps, photography, holography and laser bubble sizing. To avoid a lengthy and tedious review of bubble population literature we have summarized in table 1.1 results accumulated by various investigators since 1970. Studies had been conducted prior to 1970 [Blanchard and Woodcock, 1957; Glotov, 1962] but the instrumentation used were less reliable and accurate than post 1970 studies.

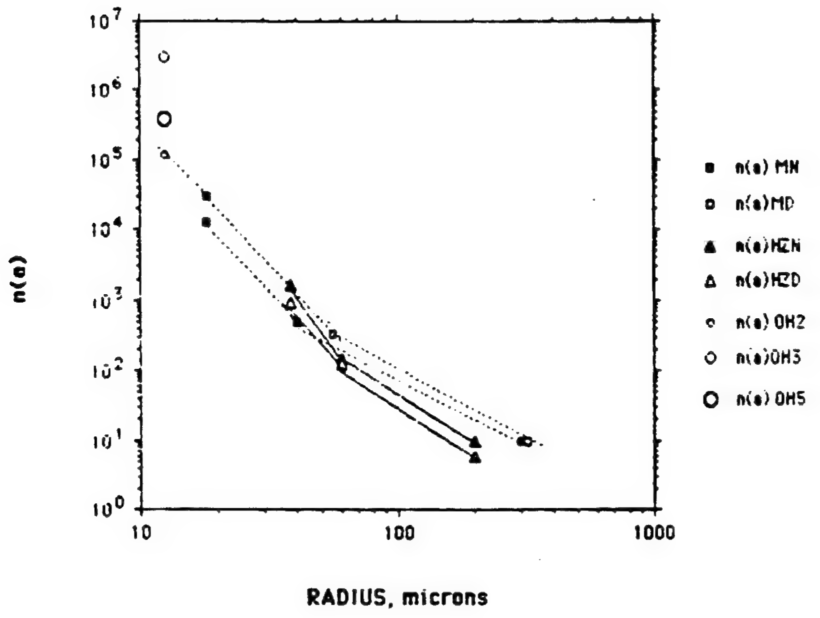


Figure 1.2: Typical bubble population measurements. From Medwin and Breitz [1989].

Although there are many discrepancies in the above investigations of bubble populations; some studies appear to agree better on issues such as the spectral slopes of the bubble population. For example, there seems to be some agreement that the slope of the population in the range 20μm<a<60μm is a^{-4} and for bubbles greater than 100μm, the slope is closer to $a^{-2.5}$. In between these two regimes, there is a transition zone anywhere from 60μm to 100μm. It should be realized that this is a personal assessment of the more consistent results which have come out of the various studies on bubble population (table 1.1). Other investigators could arrive at different conclusions depending on the merit that they assess to each study. Figure 1.2 shows typical bubble populations from several investigators as reported by Medwin and Breitz [1989]. In general, their data show an a^{-4} regime followed by an $a^{-2.5}$ regime starting at 60μm. Note that other issues such as the location of the peak of the bubble population, its magnitude, its dependence on wind, wave and depth still need further studies before generalization can be made.

Finally, it should be mentioned that a recent laboratory study by Kalvoda [1992] has started to look into the issue of the transient large bubble (greater than 1mm) population present within the first wave period after breaking. He found that half a wave period after onset of breaking, the bubble population had a slope of $a^{-3.8}$ in the range $1\text{mm} \leq a \leq 5\text{mm}$.

1.4.2 Calculation of void-fraction and sound-speed from bubble population measurements

Bubble population measurements can be integrated over all bubble sizes to yield the volume fraction of air α

$$\alpha = \int_{a} \frac{4}{3}\pi a^3 \, n(a) da \qquad (1.1)$$

where a is the bubble radius and n(a) is the number of bubbles per unit volume per unit radius. From the void-fraction, one can compute the low-frequency sound-speed using Wood's equation [Wood, 1941] as outlined in appendix A.

Since the bubble population for bubbles larger than 100μm as been found by many studies to slope between $a^{-2.5}$ and a^{-3}, is becomes clear that the integral shown in equation 1.1 diverges unless it is cutoff by an upper-bound on the size of the larger bubbles. It is also evident that with such mild bubble population slopes, most of the contribution to the void-fraction will come from the larger bubbles. However, this claim does not hold true if the slope is greater than a^{-3} and this explains why other investigators, who have used different assumptions, have arrived at the reverse conclusion that small bubbles (below 100μm) contribute more significantly to the void-fraction integral [Vagle and Farmer, 1992]. This contradiction only reemphasizes the wide discrepancies that exist across bubble population studies.

Clearly, either the bubble population cuts off sharply at some large bubble radius or it assumes a steeper slope (steeper than a^{-3}) at a certain upper-cutoff such that the void-fraction integral converges. Unfortunately, none of the field studies shown in table 1.1 have investigated this issue which is critical to the computation of the void-fraction (and sound-speed) especially if the bubble population is described by mild slopes at the upper range of bubble sizes (between a^{-2} and a^{-3}). The laboratory work by Baldy [1988] shows that the slope of the bubble population becomes very steep past a certain bubble size upper cutoff. Thus, his data confirmed the necessary existence of an upper-cutoff. His data also showed that at larger wind speeds and at shallower depths, the cutoff is pushed toward larger bubble sizes as one would intuitively expect.

Estimates of the void-fraction and sound-speed obtained using bubble population data are very sensitive to the shape of the population especially at the larger bubble sizes (say greater than 100μm). For example, assume that a bubble population ranges from 100μm up to an upper limit r_c and that the slope of the population is a^{-s} and the number density per μm increment is 1000 at a=100μm (consistent with table 1.1). We neglect in this model bubbles smaller than 100μm since, as stated earlier, we expect the larger bubbles to dominate the void-fraction integral. Using equation 1.1 to obtain the void-fraction and Wood's equation (derived in appendix A) to obtain the speed of sound, we get the results shown in figure 1.3 where the sound-speed departure from its bubble free value is plotted as a function of the upper-cutoff in the bubble population. We expect that the upper-cutoff could range anywhere from 200μm to 2000μm depending on the sea-state and the depth of the measurements. This range is consistent with Baldy's measurements in the laboratory [1988].

This simple model shows that for $s \geq -3$, the sound-speed anomaly ΔC increases more rapidly as the cutoff is pushed to higher bubble sizes. This clearly shows that ΔC is dominated by larger bubbles for these mild slopes and that the integral diverges. The most striking feature of figure 1.3 is the very high sensitivity of ΔC on s and r_c especially when s is small (say s=-2.5 to -3.5). Consequently, unless the characteristics of the bubble population for large bubble sizes is known accurately (i.e. slope and upper-cutoff), estimates of sound-speed anomalies based on bubble population data will be questionable at best.

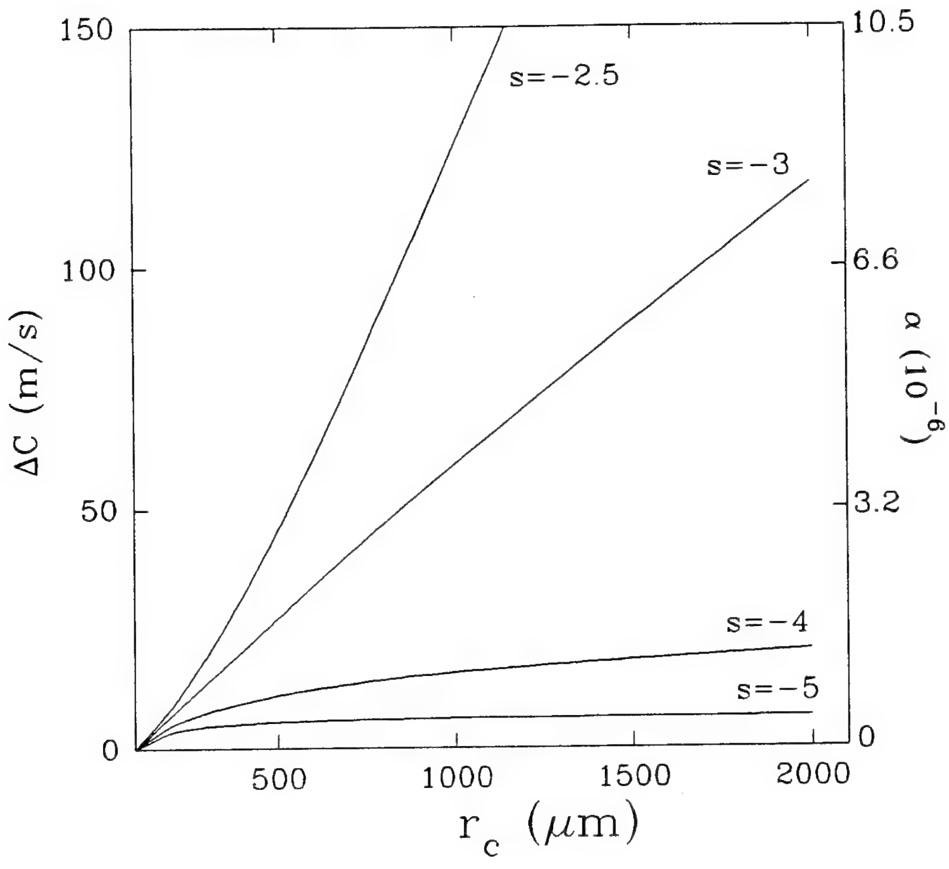


Figure 1.3: Sound-speed anomaly ΔC for a simulated bubble population with a range of bubbles from 100μm up to an upper limit r_c. The bubble population is assumed to slope at a^{-s} and the number density per μm increment is taken to be 1000 at a=100μm. Note the $\Delta C = C_w - C_m$ where C_w is the speed of sound in bubble free water and C_m is the slower speed in the bubbly mixture. The axis on the right-hand side shows the void-fraction.

Although bubble population measurements are essential for estimating gas and heat flux across the air-sea interface, it is evident that the technique is less than optimal for estimating low-frequency sound-speed near the ocean surface. A novel and more appropriate instrument for these kind of measurements is presented in §6 and actual field measurements are given in §7. The issue of determining the bubble population upper-cutoff is also addressed.

1.5 Air entrainment by breaking waves

Although bubble population measurement techniques give useful information on the quasi-steady bubble population present below the sea surface; these techniques are severely impaired when used in newly formed bubble plumes. Indeed, it is a combination of poor response time, measuring volume and dynamic range that makes these instruments inappropriate for measuring the structure of newly formed bubble plumes which evolve rapidly in time and space and which have very high initial bubble concentrations (see photographs in figure 2.3). Nevertheless, the initial wave period after the formation of a bubble plume represents the significant part of its "active" life and the large bubbles (say greater that 1mm) that are entrained during that stage only remain in the water for times shorter than a wave period [Toba et al., 1975; Su et al., 1984; Rapp and Melville, 1990; Kalvoda, 1992]. Thus, new techniques must be developed to study the evolution of bubble plumes immediately after their formation.

In this respect, modeling of the air entrainment process based on visual observation of spilling breakers [Longuet-Higgins and Turner, 1974] and on the dye-dispersion experiment of Rapp and Melville [1990] [Bell, 1989] have been attempted with moderate success mostly due to the lack of experimental measurements capable of yielding information on the internal structure and evolution of the bubbly flow generated by breaking.

Hence new instrumentation and measurements are needed to describe the flow field immediately after breaking. The former is addressed in §2 while the later, a major thrust of this work, is described in §3 and §4. We leave this topic by pointing out that measurement of quasi-steady bubble population have been conducted for nearly 30 years but the problem of quantifying the evolution of the bubble population in newly created bubble plumes is still unresolved although some recent progress has been realized [Koller et al., 1992a; Kalvoda, 1992].

1.6 Gas transfer across the air-sea interface

Due to the increases in anthropogenic 'green house' gases in the atmosphere, the study of gas transfer across the air-sea interface is more pertinent today than ever before. The ocean is known to play a crucial role in the carbon cycle by absorbing about half of the carbon from anthropogenic origins [Etcheto and Merlivat, 1988]. The global budgets of carbon dioxide and other greenhouse gases have a direct impact on the climate in general, and on global warming in particular. Climate models must take into account the dependence of the gas transfer rate on wind and other environmental parameters (i.e. wave field) to make correct predictions of atmospheric and oceanic gas concentrations [Memery and Merlivat, 1985b].

The problem of gas transfer across the air-sea interface is intimately related to the entrainment of air by breaking waves. Indeed, gas can be exchanged at the surface through turbulent diffusion and this transfer rate can be enhanced by bubbles through increases in the surface area available for exchange and increases in the ambient pressure as the bubbles are advected to greater depths[4]. It is the enhancement of gas transfer by bubbles that is of most interest to the present work.

1.6.1 The empirical transfer velocity

The precise calculation of the gas transfer across a rough air-water interface would require knowledge of the geometry of the interface, the concentration of gas and turbulence both in the water and in the air, and the bubble population below the surface. Clearly these measurements are very difficult to make and they are presently not possible over large areas (~km^2 or more). The easier approach which has been followed to estimate gas transfer across an air-water interface is to use an empirical formulation where the flux of gas is given by the product of the gas concentration difference across the interface and the transfer velocity (or piston velocity) specific to that gas. With this formulation, the transfer velocity is therefore an empirical function.

Estimates of the transfer velocity have been obtained from bomb-produced ^{14}C [Broecker and Peng, 1974, 1984] and from the radon deficit method [Roether and Kromer, 1984; Roether, 1986]. Both isotopic methods involve large spatial (~100km) and temporal (~several days) averages and therefore they fail to describe the gas flux dependence for scales more appropriate to a storm. Fluxes calculated using the eddy correlation technique ($F \sim \langle c'w' \rangle$ where c is the gas concentration and w is the vertical wind velocity) [Smith and Jones, 1985; Wesely at al., 1982] have been seriously questioned because they showed fluxes two orders of magnitude greater than isotopic fluxes [Broecker et al., 1986]. It was proposed that these discrepancies could be explained by the important differences in temporal and spatial scales between the two methods [Smith and Jones, 1986].

Several studies in wind-wave tunnels [see review by Liss and Merlivat, 1986], in lakes [Wanninkhof et al. 1985] and in the ocean [Watson et al., 1991] have made direct measurements of the empirical transfer velocity. Three distinct regimes define the relation between transfer velocity and wind speed. The smooth surface regime

[4] Note that the ambient pressure doubles in the first ten meters.

(U<5m/s), the rough surface regime (U>5m/s) and the breaking-wave regime (U>10m/s) for which the transfer velocity has the greatest increase with wind speed. These studies argued that the higher transfer velocities at high wind speed were caused by bubbles entrained by breaking waves and several models supported this claim (see §1.6.2 for a discussion of the models). However, a recent study by Khoo and Sonin [1992] has demonstrated that similar regimes can be obtained with only near-surface turbulence without any bubble entrainment. Hence, their study suggested an alternate and most likely complementary mechanism for the enhancement of gas transfer.

It should also be pointed out that experiments in large wind-wave tanks [Jahne et a., 1985, 1987; Merlivat and Memery, 1983] have shown that the transfer velocity is not only dependent on the wind speed but also on the wave field and thus, it may not be straightforward to apply wind tunnel transfer velocities to the open ocean where wave conditions can be significantly different.

1.6.2 The enhancement of gas transfer by bubbles

The contribution of bubbles to the total gas flux has been modeled [Merlivat and Memery, 1983, Memery and Merlivat, 1985a] and studied in laboratory wind-wave tunnels [Broecker and Siems, 1984; Merlivat and Memery, 1983]. These studies have shown that bubbles generated by breaking waves significantly enhanced gas transfer (80% for Ar and 60% for N_2) and that the enhancement effect started at lower wind speeds for less soluble gases (e.g. O_2 is less soluble than CO_2). Furthermore, supersaturation of dissolved gas in the water was found to be the normal condition when there is equilibrium in the gas flux between the atmosphere and the ocean. Supersaturation is caused by an increase in bubble gas flux generated by greater pressure inside the bubble which are caused by surface tension and hydrostatic pressure. Models of the contribution of bubbles to gas transfer based on field measurement of bubble population have also obtained the same general findings as those described above [Thorpe, 1982; Woolf and Thorpe, 1991]. Finally, we point out that results from two recent field experiments on the measurement of long-term time-series of dissolved oxygen at sea showed sudden large increases associated with surface wave activity followed by long degassing period during calmer conditions [Wallace and Wirick, 1992; Farmer et al. 1993]. These studies showed the importance of gas transfer on the spatial and temporal scales of a storm.

The total gas flux across a bubble is controlled by four time scales [Jahne et al. 1984]: 1) the e-folding time T_g for the gas in the bubble to come to equilibrium with the dissolved gas in the water; 2) the bubble residence time in the water T_z which is mainly dependent on the depth of the bubble and its rise velocity; 3) the e-folding time involved with the increased gas transfer caused by hydrostatic pressure and 4) surface tension. Figure 1.4 shows the equilibrium times T_g and T_z as a function of bubble size for different gases. In general, T_g increases with bubble radius and decreases with gas solubility. Hence, small bubbles reach a gas concentration equilibrium before they reach the surface and large bubbles reach the surface before they equilibrate. The effect of hydrostatic pressure and surface tension have comparatively slower equilibrium time scales and hence they have not been included in figure 1.4 (they affect mostly small bubbles).

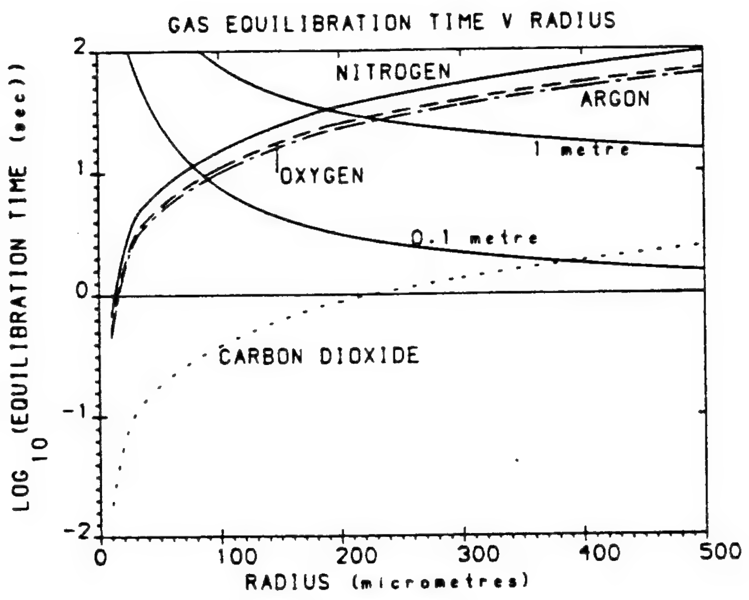


Figure 1.4: Equilibrium time scales of four gases in bubbles and the time for bubbles to rise 0.1m and 1m. From Woolf and Thorpe [1991].

Figure 1.4 indicates that the time scales for gas exchange vary significantly depending on the nature of the gas. For example, carbon dioxide contained in a bubble with a 0.5mm radius will reach its e-folding gas-transfer time in approximately 2sec. This is an important observation which shows that even large bubbles, which contain significantly more gas, can transfer their content of highly soluble gases such as carbon dioxide in a very short time, comparable to their residence time in the water.

All current models of gas transfer enhancement by bubbles use as input a quasi-steady bubble population obtained from either field or laboratory measurement of bubble densities. Unfortunately, these bubble populations are not representative of the transient population found under breaking waves, which are yet to be fully investigated. It is clear however that these transient bubble population will contain larger bubbles which typically have a residence time in the water of a second or less. It may therefore be that the neglect of these large transient bubbles is a serious omission in present gas transfer models. This issue is further explored in §3.

1.7 Ambient sound in the ocean

In general, the ambient sound level in the ocean increases with wind speed and decreases at approximately 5 to 6dB/octave over the frequency from 100Hz to 25kHz [Knudsen et al., 1948; Wenz, 1962; and for a relatively recent review of ambient sound in the ocean, see Urick, 1984]. Figure 1.5 shows a composite spectrum of the sound pressure level as a function of frequency. The curves labeled 0, 2, 4 and 7 show how the ambient sound increases with wind force (Beaufort scale).

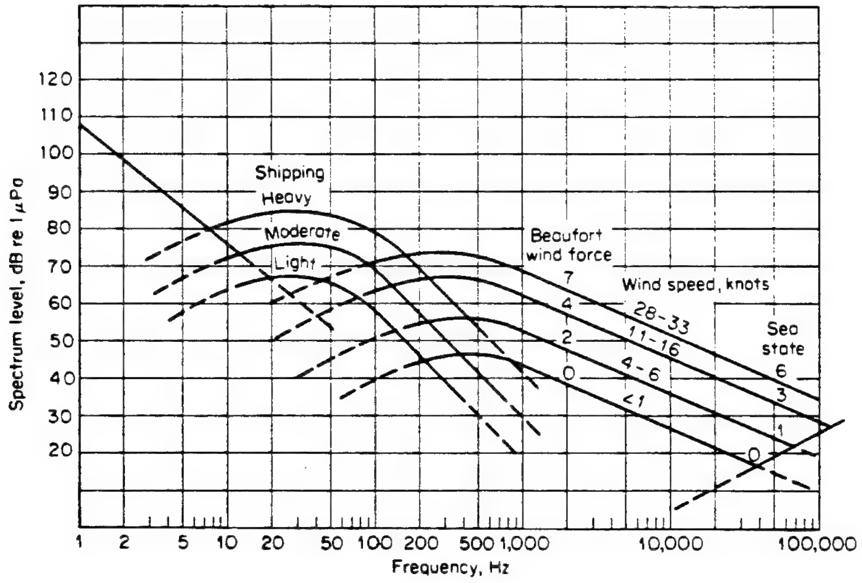
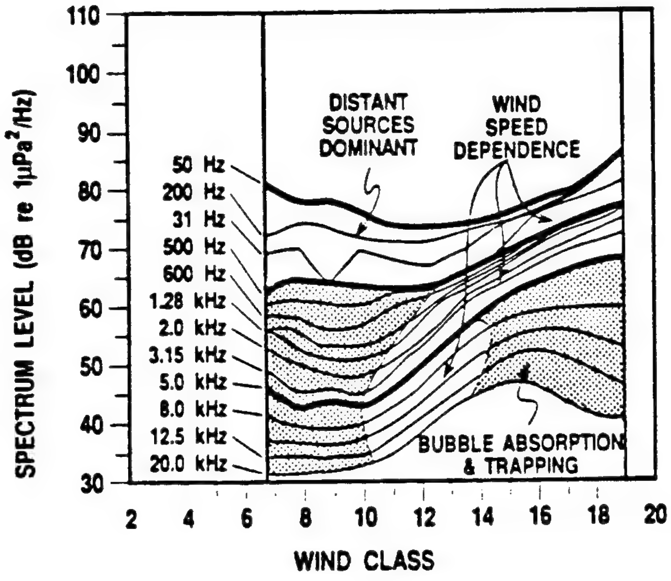


Figure 1.5: Average deep-water ambient-noise spectra. From Urick [1975].

It was also observed that for higher frequencies (say above 10kHz) the sound spectrum level does not increase monotonically with wind speed. Instead, it was found to decrease at high enough wind speeds (12m/s and above) [Farmer and Lemon, 1984; Carey et al., 1993]. This effect is shown in figure 1.6 where a composite plot of the ambient sound spectrum level is plotted as a function of wind speed [Carey et al., 1993]. The attenuation at higher frequencies was attributed to the presence of a dense bubbly layer which is formed close to the surface at high wind speeds.



WIND CLASS	WIND SPEED
7	1.6 - 2m/s
8	2 - 2.5m/s
9	2.5 - 3.15
10	3.15 - 4
11	4 - 5
12	5 - 6.25
13	6.25 - 8
14	8 - 10
15	10 - 12.5
16	12.5 - 16
17	16 - 20
18	20 - 25
19	25 - 31.5
20	31.5 - 40

Figure 1.6: Spectrum level of ambient sound as a function of wind speed and frequency. From Carey et al. [1993].

Several mechanisms have been suggested for explaining the wind dependence of ambient sound in the ocean and the reader is referred to Kerman [1988, 1993] for recent reviews of the various mechanisms presented. It is now generally accepted that breaking waves are the dominant source of natural ambient sound in the sea at frequencies from ~500Hz up to 25kHz [Kerman, 1984; Farmer and Vagle, 1988; Pumphrey and Ffowcs Williams, 1990; Crowther, 1988]; and that they can also radiate significant sound levels at frequencies down to 10Hz [Farmer and Vagle, 1989; Hollet, 1993; Loewen and Melville, 1993]. Figure 1.7 shows a spectrum of the sound pressure level for laboratory breaking waves. The various curves which increase upward represent 4 different breaking wave intensities. Clearly, significant amount of sound is generated at the low frequencies down to 10Hz.

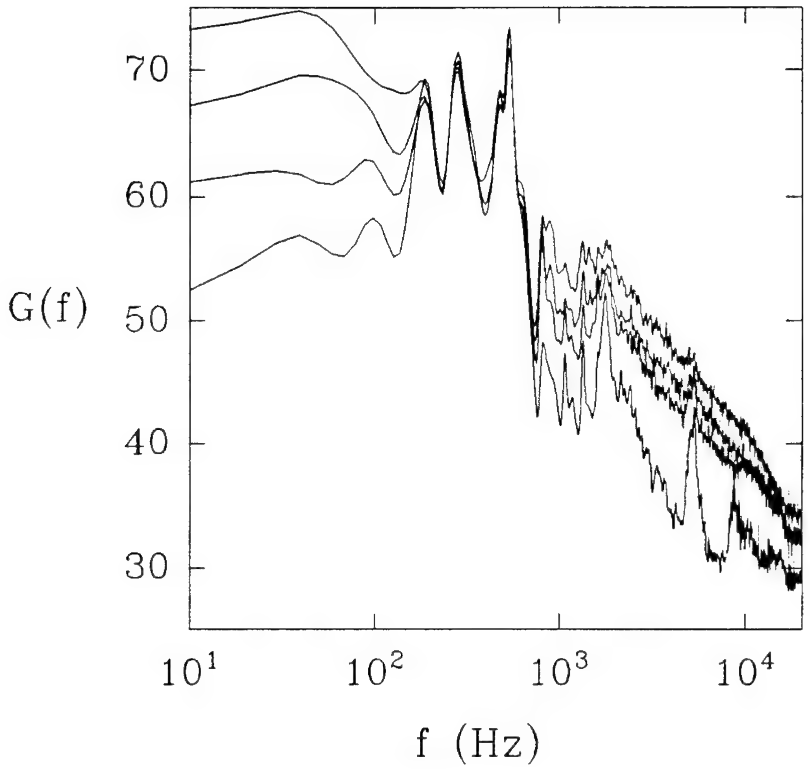


Figure 1.7: Pressure spectrum level G(f) in dB re 1μPa^2/Hz of the hydrophone signal band pass filtered between 10Hz and 20kHz and band reject filtered from 200Hz to 340Hz. Each spectrum is the average of five repeats of the breaking wave in the Texas A&M 3D wave basin. The four different curves correspond to four different amplitudes of the breaking wave packet. The lowest curve is the incipient case (i.e. no breaking). Note that the sound in the 200Hz to 800Hz band is corrupted by hydraulic noise from the wavemaker system. From Loewen and Melville [1993].

Although it is clear that breaking waves generate sound over the full audio spectrum and beyond, the actual mechanisms responsible for this sound within the breaking waves remains to be determined. A consensus is building, among ocean acousticians, supporting two dominant mechanisms for the generation of sound at high-frequencies (above 500Hz) and at low frequencies (below 500Hz). These two mechanisms are reviewed below.

1.7.1 Bubbles as a source of high-frequency sound

At high-frequencies (500Hz to 20kHz), there is now a wide agreement that the wind-dependent ambient sound is due primarily to individual bubbles. Indeed, laboratory experiments [Banner and Cato, 1988; Medwin and Beaky, 1989; Medwin and Daniel, 1990; Pumphrey and Ffowcs Williams, 1990], field experiments [Updegraff and Anderson, 1991] and modeling [Loewen and Melville, 1991b] have shown that the primary source of sound in breaking waves at high-frequencies is newly created bubbles oscillating at their lowest linear resonant frequency (the "breathing" mode).

1.7.2 Bubble plumes as a source of low-frequency sound

At low frequencies (below 500Hz) it has been proposed that the collective oscillations (or volume oscillations) of bubble plumes generated by breaking waves could generate significant sound levels [Prosperetti, 1988; Carey and Browning, 1988]. The essential idea behind this hypothesis is that a compact bubbly region composed of small bubbles can radiate sound as a monopole source similar to the oscillations of an individual air bubble. Although the boundaries of the plume are not as well defined as that of a bubble, the plume is nevertheless localized and compact with a distinctively lower compressibility given by its constituent bubbles and an inertia from the mass of the plume itself. Consequently, as in the case of an air bubble in water, the plume can experience harmonic oscillations when excited by an external pressure forcing, and it should come as no surprise that the frequency of such oscillations is described by a modified Minnaert formula [Minneart, 1933; see his original formula in appendix B, equation B.2]

$$f_o = \frac{1}{2\pi\, r_o}\sqrt{\frac{3\gamma P}{\rho_w\, \alpha}}$$

where r_o is the radius of the spherical bubble plume, ρ_w is the density of water, P is the ambient pressure, γ=1.0 is the specific heat for isothermal conditions and α is the void-fraction assumed constant throughout the plume [Carey and Browning, 1988]. This result and other similar formulae differing only by numerical constants, were obtained by several investigators who used different modeling approaches [d'Agostino and Brennen, 1983, 1988; Omta, 1987; Carey and Browning, 1988]. Although the above formulation is for a spherical bubble plume in an infinite ocean, similar formulations can also be derived for different geometries [Carey et al., 1993]; and for plumes located close to a pressure release surface where they will radiate sound as a dipole source [Loewen and Melville, 1993]. Figure 1.8 shows the predicted resonant frequencies of a spherical and cylindrical bubble plume as a function of the void-fraction and radius. We find that for realistic values of void-fractions and bubble plume radius (as will be shown in the present work), the resonant frequencies of the bubble plume are very low.

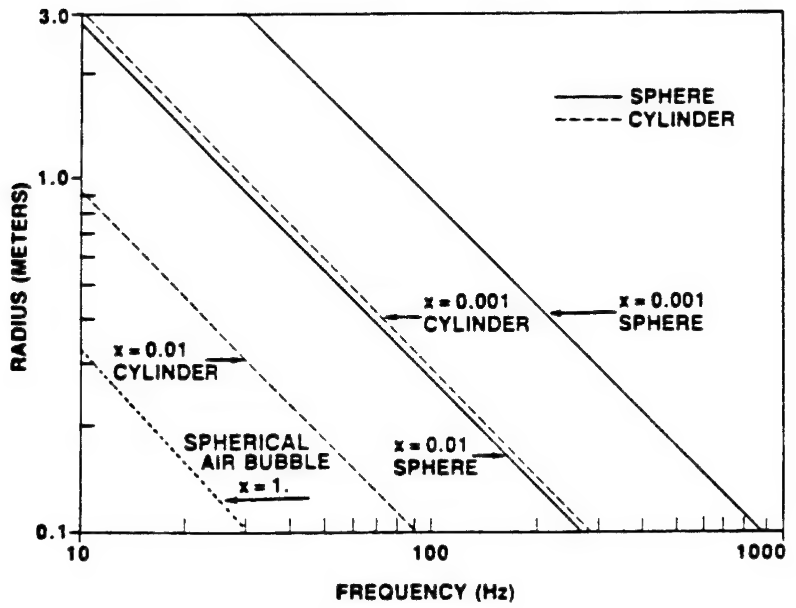


Figure 1.8: Predicted resonant frequencies of a spherical and cylindrical bubble cloud as a function of the void-fraction and plume radius. From Carey et al. [1993].

Lu et al. [1990] have derived a linear model for the resonant frequencies and the attenuation of a bubble plume. They considered free oscillations as opposed to forced oscillations in the case of Carey and Browning [1988] and included damping. Their model also allowed for the determination of all eigenfrequencies, not only the fundamental one. Although the model permits a greater insight into the dynamics of the bubble plume oscillations, its predictions of the fundamental frequency of oscillation are consistent with the simpler model by Carey and Browning [1988]. There is therefore strong evidence from modeling that bubble plumes can undergo volume oscillations, and some of the recent experimental evidences, which are reviewed below, support the theory.

Yoon et al. [1991] have conducted a laboratory experiment in which a cylindrical bubble column was generated by a series of hypodermic needles located at the bottom of a tank. The peak low frequency acoustic signature from the bubble column (260Hz to 550Hz) matched very well the theoretical frequencies predicted by the bubble cloud model of Lu et al. [1990]. Similar modeling and measurements by Koller and Shankar [1993] have shown that the pressure field outside the bubbly mixture was evanescent instead of oscillatory.

Kolaini et al. [1991] have conducted experiments on the low frequency acoustic signature of bubble plumes generated by dropping water from a cylindrical container. They observed that low frequency sound was generated when a large "substructure" detached from the plume. Their observations suggested that the substructures were spherical regions of high void-fraction and they found that the lowest resonance frequency would match the measurements if the mean void-fraction was assumed to be approximately 40%.

Experiments conducted by Roy et al. [1992] on the scattering of deep submerged bubble clouds generated in a lake have shown that the cloud of bubbles responded to the pressure field excitation according to the theory of low-frequency volume oscillations (as described above).

Finally, recent measurements by Carey et al. [1993] on the sound generated by a tipping trough in both fresh and salt water have shown that the bubble plume generated by the tipping trough radiated sound at low-frequencies consistent with theory. They also found that the frequency of oscillation did not change between fresh and salt water which suggests that the plume had similar dimensions and void-fractions in both cases although it was clear from their measurements that the bubble population was different as is expected for fresh and salt water. The main difference observed between fresh and salt water was in the level of the acoustic radiation where it was found that the salt water case had lower levels due to the higher proportions of small bubbles which attenuate more strongly the low-frequency oscillations.

All of the above investigations have demonstrated that plumes of bubbles generated by artificial methods can undergo low-frequency volume oscillations. The major deficiencies of these studies are the lack of geometrical similarity of the bubble plumes to those generated by breaking waves in the ocean; and the lack of measurements of the void-fraction field within the plume. This last comment does not apply to the experiment of Yoon et al. [1991] where void-fraction measurement were estimated from air-flow rate measurements and bubble rise time. Indeed, most investigators have estimated the size of and the void-fraction within their plumes and consequently, only rough estimates of the resonance frequencies can be made.

Thus, progress has been made in identifying the possible mechanism for the generation of low-frequency sound (below 500Hz) in the ocean but it remains to be shown that volume oscillations exist in bubble plumes generated by breaking waves. An accurate validation of this hypothesis will only be possible when the space-time evolution of the void-fraction field under realistic breaking waves is measured along with measurements of the sound generated. This issue is addressed in §3 and §4 of this work.

1.8 Measurement of sound-speed anomalies and attenuation in the surface layer

As it was outlined in §1.2, sound is a powerful tool which can be used to remotely monitor surface processes such as breaking waves, surface wind speeds, rainfall rates and maybe, in the near future, energy and mass transfer across the air-sea interface. However, many of the benefits offered by acoustic remote sensing depend heavily on the prior knowledge of the propagation characteristics between the source (i.e. the phenomenon being observed) and the receiver located at some depth below the surface. Without that information, it would be difficult to separate acoustic features that are caused by the source from those that are caused by the medium through which the sound must travel. Consequently, there is a need to study the propagation characteristics of the upper-ocean. A suitable description would include spatial and temporal variations in the sound velocity and its attenuation.

1.8.1 Measurements of sound-speed anomalies

Only a few previous studies have been conducted on the measurement of sound velocity at sea [Medwin 1974, Medwin et al. 1975a, Farmer and Vagle 1989]. These investigations deserve special attention considering the direct relevance they have on the present work. We shall therefore closely examine their measurement technique and their results.

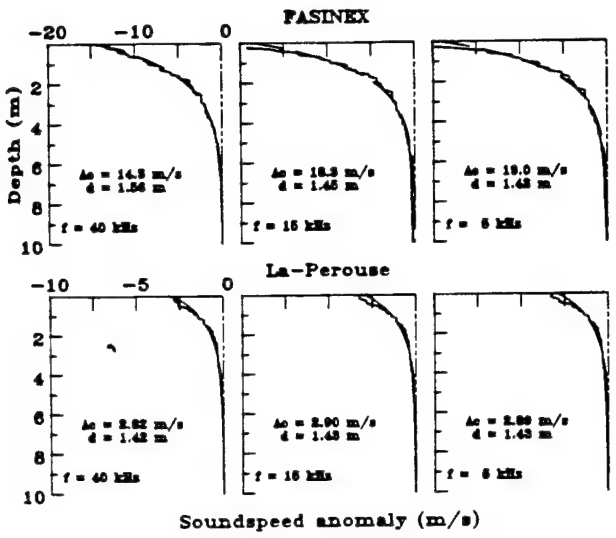


Figure 1.9: Time-averaged (15min) profiles of the sound-speed anomaly $\Delta C = C_m - C_w$ as a function of depth for two different experiments (Fasinex, 12m/s winds and La Perouse, 10m/s winds). Note that C_w and C_m are the speed of sound in bubble-free water and in the bubbly mixture respectively. Three frequencies 5, 15 and 40kHz are shown. The e-folding depth of the profile is d and the anomaly at the surface ΔC are written on each plot. From Farmer and Vagle [1989].

As demonstrated in §1.4, one can compute the low-frequency sound-speed of a bubbly mixture by integrating the bubble population over the full spectrum of bubble sizes present in the mixture (equation 1.1). It is also possible to remove the low-frequency restriction by adding to the integral the dispersive effects caused by individual bubbles excited close to or at their resonance frequencies (this modified integral equation is given in appendix B). Farmer and Vagle [1989] have inferred time-averaged sound-speed profiles in the upper-ocean by integrating their bubble population measurements and by including possible contributions caused by bubble dispersion (figure 1.9). Their data shows that the sound-speed anomalies are the largest at the surface (3 to 15m/s) and decrease exponentially with depth with an e-folding depth of ~1.4m. It is well known that sound-speed profiles with sound velocities increasing with depth can act as a waveguide where sound may be trapped between the surface and the upward-refracting profile. Also notice that Farmer and Vagle [1989] have observed very little dispersion up to 40kHz suggesting that dispersive effects caused by bubbles were negligible up to those frequencies.

Farmer and Vagle [1989] also made measurements of the ambient sound generated by breaking waves. The spectrogaphs from three such breaking events are shown in figure 1.10. The remarkable features of these three independent spectrographs are the distinct peaks in the sound spectrum which were found to be repeatable from one breaking event to the next but varied significantly from one storm to another. These spectrographs gave an excellent example of the kind of acoustic filtering that can be caused by the medium through which the sound must propagate. Models of sound propagation through an exponentially decreasing sound-speed profile [Farmer and Vagle, 1989] and an inverse square profile [Buckingham, 1991] demonstrated that the features of the sound spectrum were caused by the waveguide.

Farmer and Vagle [1989] determined their bubble population by measuring their number density at four discrete radii (16, 37, 65 and 116μm) at depths ranging from 0.1m to 10m with an inverted echo sounder operating at four discrete frequencies: 28, 50, 88 and 200kHz. They computed the sound-speed by integrating the bubble population. They found the peak of their bubble population to be consistently located between 16 and 37μm but the exact position could not be established. The slope of the bubble population to the left (smaller radii) of the peak was assumed to be a^3 and therefore bubbles in that region had negligible contribution to the sound-speed integral equation. For the bubbles to the right of the peak, they found slopes between a^{-4} and a^{-6} which is consistent with the literature for bubbles up to 60μm but is not consistent for larger bubbles which have been found by many investigators to have an $a^{-2.5}$ slope up to about 270μm which was the limit of their instrumentation [Medwin, 1970,1977a; 1988; Medwin and Breitz, 1989]. Note that Su et al. [1993] have measured an a^{-3} slope up to 1200μm but criticism was expressed in §1.4 as to the problems associated with not knowing exactly the location of the upper-cutoff in bubble size. Farmer and Vagle's [1989] instrumentation could not measure the bubble population density for radii above 116μm and therefore it may have been difficult for them to detect the change in slope between 60μm to 100μm previously observed. Nevertheless, the slope of the distribution above 60μm to 100μm is an important issue which deserves further examination.

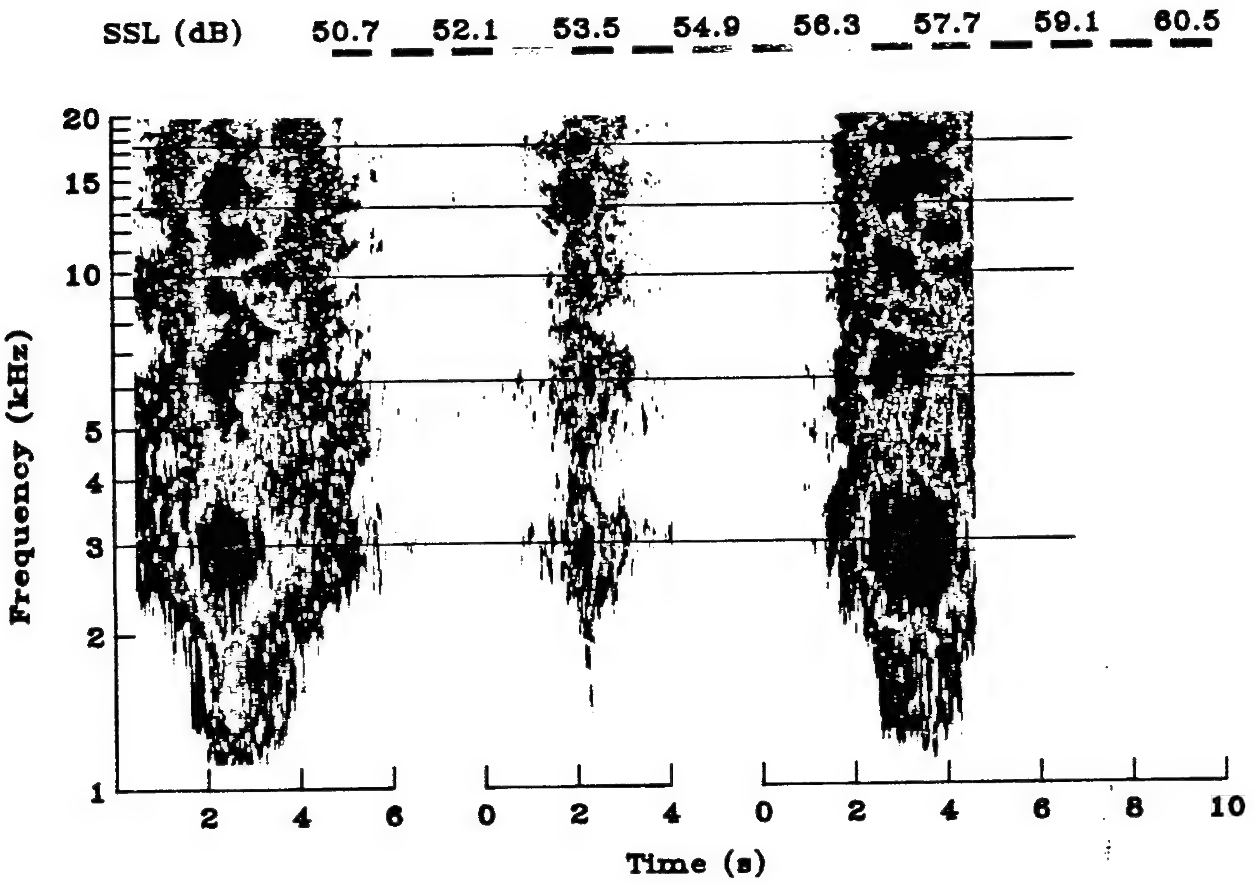


Figure 1.10: Spectrographs of the sound generated by three separate breaking events (Fasinex experiment). The horizontal lines correspond to the cutoff frequencies predicted by Farmer and Vagle's [1989] model. From Farmer and Vagle [1989].

We can appreciate the effect of a different slope above say 60μm by computing the low frequency sound-speed with a bubble population having the same characteristics as described above but with an a^{-4} slope in one case and an $a^{-2.5}$ slope in the other. The number density of bubbles at the peak of the population and at the surface is obtained from Vagle's thesis [1989] and the computation is carried from 1μm up to a maximum radius of 400μm. The sound-speed reduction with an a^{-4} slope is found to be 26.8m/s ($\alpha \sim 1.65 \times 10^{-6}$, where α is the void-fraction) which is comparable to Farmer and Vagle's [1989] computations at the surface for their lowest frequency (figure 1.9), and the reduction with an $a^{-2.5}$ slope is 97.2m/s ($\alpha \sim 6.44 \times 10^{-6}$), a factor of 3.6 greater. A similar kind of calculation with the peak of the population varying between 16μm to 24μm shows that the sound-speed reduction would vary by a factor of two. Thus, it is again clear that sound-speed calculations based on integration of the measured bubble population are very sensitive to the exact shape of the bubble population for which there are still large discrepancies in the literature. In the case of Farmer and Vagle's data, combined errors on both the slope and the peak of the population could sum up to an order of magnitude difference between the actual and the calculated sound-speed.

We now move on to a brief review of measurements of sound-speed fluctuations near the surface by Medwin [1974] and Medwin et al. [1975a]. Their experiments consisted of measuring the phase shift of a pure tone CW signal between two horizontal hydrophones at various frequencies in the range 15 to 100kHz. The sound-speed was computed by measuring phase shifts at sea and by comparing them to reference phase shift in bubble free water. Their instrument had a high-resolution of 0.1m/s but it was limited to a minimum measurement depth of 3m in order to minimized surface reflections which would corrupt the phase. Their results are shown in figure 1.11. Maximum sound-speed anomalies on the order of 10 to 15m/s were observed at depths greater than 3m during very light wind conditions. This would suggest that much higher values are likely to be found close to the surface at higher sea-state. They argued that the sound-speed was no longer dispersive below 25kHz but their data is limited and too scattered in that lower frequency range to fully support this claim. The peaks observed at 60kHz and 70kHz were attributed to peaks in the bubble population.

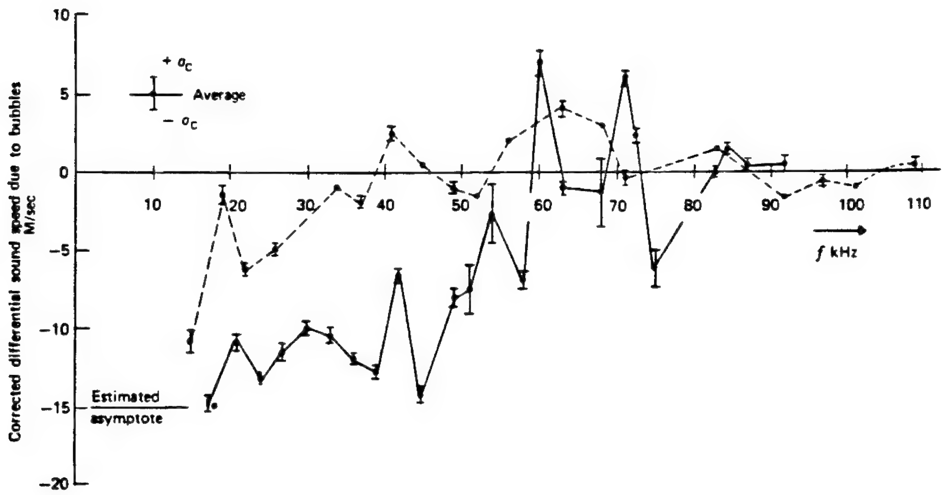


Figure 1.11: Sound-speed anomalies as a function of frequency including standard deviation bars. Dashed line taken at 3.3m below the surface in water depth of 15m and wind speed of 6 knots. Solid line data obtained 5m below the surface adjacent to an ocean tower during wind speeds of 8 to 10 knots. From Clay and Medwin [1977].

Medwin's measurement of sound-speed at sea were limited to a minimum depth of 3m while Farmer and Vagle's measurements were mostly impeded by the difficulties in computing the sound velocity from bubble population data and by the limited dynamic range of echo-sounders at high void-fraction. It therefore appears that other measurement techniques must be developed to circumvent these limitations and permit accurate measurement of sound velocities close to the surface. This is the thrust of §6 and §7.

1.8.2 Measurements of attenuation

The surface layer not only affects the propagation speed of acoustic waves but also their attenuation. Measurements have been conducted in the North Sea at depth of ~1.5m and greater on the acoustic attenuation in the frequency range 3kHz to 100kHz [Herwig and Nutzel, 1989]. The attenuation was measured along a fixed 2.4m path separating a transmitter and a receiver located at the same depth. The results from that study showed that attenuation increased with sea-state and frequency and decreased with depth. More importantly, they observed significant short-term fluctuations which dominated the long-term average. Figure 1.12 shows a 4min time-series of the attenuation at a depth of 3.1m in very strong wind conditions. The high fluctuations up to 30dB/m are believed to be from bubble clouds drifting by the sensor. The results clearly demonstrate that 10 or 20min averages may not be meaningful in the presence of such high fluctuations. The measurements also suggest that similar fluctuations may be present in time-series of sound-speed. This issue is addressed in §7.

1.9 Outline of thesis

From the discussion above, it may be clear that while we have some understanding of long-term upper-ocean characteristics such has bubble population, gas transfer and sound-speed, very little is known of their transient nature which, in many instances, may dominate and control the long-term behavior.

The origin of air entrainment at the ocean surface is undoubtedly wave breaking which initially generates very dense bubble plumes. With time, the large bubbles within these plumes rise back to the surface leaving behind a cloud of fine bubbles with weak buoyancy and hence very slow rise velocities. These freely drifting clouds eventually mix with the background bubble densities or organize in Langmuir cells.

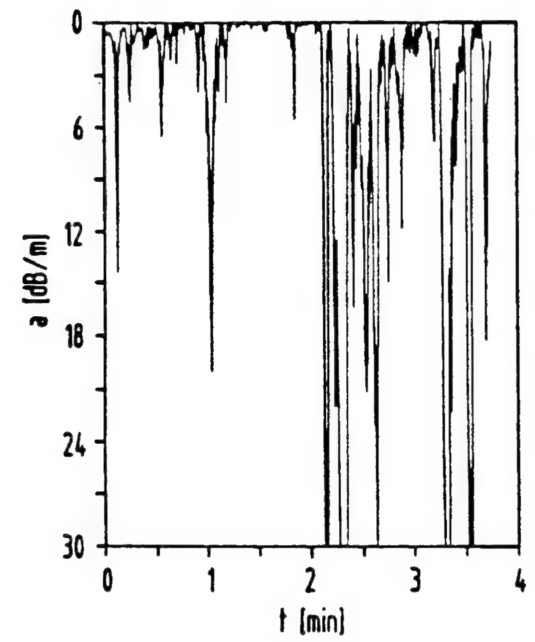


Figure 1.12: Time history of measured attenuation at a frequency of 30kHz, wind speed of 17.5m/s and sensor depth of 3.1m. Note that the separation between the transmitter and the receiver was 2.4m. From Herwig and Nutzel [1989].

The initial entrainment of air which occurs a few seconds after the onset of breaking accounts for the most rapid and significant part of the plume/cloud evolution and, unfortunately, it is the least studied and consequently the least well known. This is almost entirely due to the difficulties inherent in modeling these kinds of flows and, most importantly, to the lack on instrumentation capable of yielding valuable information on the internal structure and evolution of the plume.

The present study is devided in two principal sections. The first one, on the measurement of void-fraction, comprises §2 to §5 which address the issues outlined above. In Chapter 2, we present the development of novel laboratory and field instrumentation for the measurement of void-fraction in bubble plumes generated by breaking; and we demonstrate its capabilities in the laboratory by mapping the complete evolution of the plume's void-fraction field from onset of breaking up to a full wave period after breaking. In Chapter 3 and Chapter 4, we present an extensive set of measurements of air entrainment in 2D and 3D laboratory breaking waves. In particular, we show that the void-fraction measurement can yield several important bulk properties of the bubble plume and their evolution in time. These measurements place a significant emphasis on the importance of the initial period after breaking for modeling the contribution of bubbles to gas transfer. The measurements also demonstrate that the volume oscillation of bubble plumes is a significant mechanism for the generation of sound by breaking waves. In Chapter 5, we present void-fraction measurements and video photography of bubble plumes in the field. We find the void-fraction measurements to be consistent with the laboratory measurements and the video photography reveals the presence of large bubbles which were previously undocumented. Furthermore, the void-fraction instrumentation is introduced as a new instrument for the detection of breaking waves in the ocean.

The second section of this thesis focuses on the measurement of sound-speed and attenuation. In that respect, it complements the first section by providing measurements of very small void-fractions which are outside the capabilities of the technique presented in the first section. Moreover, it expands the scope of this thesis in the domain of ocean acoustics by introducing new instrumentation for the study of sound propagation near the sea surface and by providing new measurements of the acoustic properties of the surface layer. Hence, in Chapter 6 we present the development of a new technique for measuring the sound-speed and sound attenuation profiles at low-frequencies, close to the ocean surface and at a sufficiently high sampling rate to successfully describe their fluctuations. In Chapter 7, results from two field experiments are presented on the measurement of sound-speed and attenuation. In particular, the dependencies of the measurements on frequency, depth and wind speed are studied along with their relationship to the bubble population. Finally, in chapter 8, the results of the thesis are summarized.

SECTION I

MEASUREMENTS OF VOID-FRACTION

Chapter 2

Instrumentation for the measurement of void-fraction in breaking waves

In this chapter, we report on the development of a probe to measure the volume fraction of air (hereafter void-fraction) in bubble plumes generated by breaking waves. Some of the work reported in this chapter first appeared in Lamarre and Melville [1992]. The void-fraction gauge described here is found to be most useful in the initial period after breaking when high void-fractions (O(1%) or greater) prevail. The technique makes use of the difference in electrical conductivity between air and water to yield the volumetric fraction of air. The technique is not novel but its application to unsteady free surface two-phase flows is new. This chapter gives a thorough account of the instrumentation, its calibration and various performance criteria. Actual measurements of void-fraction in laboratory breaking waves and in the field are reported in §3, §4 and §5.

2.1 Theory for the determination of void-fraction by the impedance method

Maxwell [1892, article 314] derived an expression for the effective conductivity σ_{eff} of a heterogeneous medium composed of non-interacting spheres of conductivity σ_1 dispersed in a medium of conductivity σ_2:

$$\sigma_{eff} = \frac{\sigma_1 + 2\sigma_2 + 2\alpha(\sigma_1 - \sigma_2)}{\sigma_1 + 2\sigma_2 - \alpha(\sigma_1 - \sigma_2)}\,\sigma_2\,, \qquad (2.1)$$

where α is the volumetric fraction occupied by the spheres. For a mixture of air bubbles in water, the conductivity of the dispersed phase is much smaller than the conductivity of the medium and Maxwell's expression reduces to

$$\sigma_{eff} = \frac{1 - \alpha}{1 + \alpha/2}\,\sigma_w\,, \qquad (2.2)$$

where σ_w is the conductivity of water. Maxwell's expression for σ_{eff} can be expanded to describe the effective permittivity[1], ε_{eff}, of the mixture

$$\varepsilon_{eff} = \frac{1 - \alpha}{1 + \alpha/2}\,\varepsilon_w \qquad (2.3)$$

where ε_w is the permittivity of water. We have used in this last expression the fact that the permittivity of air is much smaller than the permittivity of water (by a factor of 80).

The impedance across two electrodes immersed in an air-water mixture can be modelled as a resistance, R, and a capacitance, C, in parallel

$$R = \frac{K}{\sigma_{eff}}, \quad C = \frac{\varepsilon_{eff}}{K} \qquad (2.4)$$

where K is the cell constant (with dimensions of length^{-1}) which is a function of the geometry and spacing of the electrodes. Because of fringing effects, the cell constant is often difficult to determine when working with finite size electrodes. Fringing occurs at the edges of the electrodes where the electric field is not uniform [Zahn, 1979, pp. 173]. Using (2.2), (2.3) and (2.4), the total impedance for the parallel RC model can be written as

$$Z = \frac{K\,(1 + \alpha/2)}{1 - \alpha}\left(\frac{1}{\sigma_w} - \frac{j}{2\pi\, f\, \varepsilon_w}\right), \qquad (2.5)$$

where f is the electrode excitation frequency and $j=\sqrt{-1}$. The first term and second term in brackets are the real (resistive) and imaginary (capacitive) part of the impedance respectively. Typical values for σ_w and ε_w for fresh water are 0.005 S/m and $7.1\text{x}10^{-10}$ F/m [Olsen, 1967]. Therefore for $f \ll$ 1MHz the fluid cell becomes resistive and for $f \gg$ 1MHz it becomes capacitive. With an appropriate circuitry to measure resistance or capacitance, either regime could theoretically be used to measure void fraction but in practice it has been found difficult to eliminate capacitive pick-up and electromagnetic radiation when operating in the capacitive mode [Rubesch, 1990].

[1] The permittivity ε is also referred to as the dielectric constant. The permittivity of free space is ε_o=8.9×10^{-12} F/m and that of water is $80\varepsilon_o$. The permittivity expresses the degree of polorizability of materials subject to an electric field. In general, high permittivity materials are composed of highly polar molecules [Zahn, 1979].

When the fluid cell is excited by a direct current, the ions of the electrolyte accumulate on the solid electrodes and create an electromotive force acting in the opposite direction. This ionization of the water in the vicinity of the electrodes is known as polarization and it results in a diminution of current or an apparent increase in the resistance [Maxwell, 1892, article 264]. It is possible to significantly reduce the parasitic resistance due to polarization by exciting the fluid cell with an alternating current. The optimum frequency of excitation is a function of the composition of the electrodes and of the electrolyte. Olsen [1967] has found that polarization was eliminated with frequencies as low as 600Hz for stainless steel electrodes immersed in fresh water. The polarization effect can be detected by monitoring the drift of the output when the electrodes are immersed in bubble free water kept at constant temperature. If polarization is significant, it will show up as an apparent increase in the resistance of the fluid cell and will be detectable as drift at the output. A good example of this effect is an ohmmeter connected to two electrodes immersed in water. The meter will show immediate, steady increase in resistance until saturation of the output is reached.

2.2 Void-fraction instrumentation

Various electrode configurations were tested in order to find out the most suitable probe for void-fraction measurement in two-dimensional laboratory breaking waves. Our aim was to design a probe with minimum flow perturbation, temperature effect, surface effect and bubble-size effect. Temperature effect is caused by fluctuations in the impedance of the bubbly mixture caused by fluctuations in temperature. Surface effect is due to the influence of the water surface on the electric field and it restricts the ability of a probe to measure void-fraction close to the water surface. Bubble-size effect is due to the dependence of the instrument void-fraction output on bubble size. These effects will be discussed at length subsequently. The final choice of electrode configuration is shown in figure 2.1. It consists of three parallel nichrome wires (0.127mm diameter), 20cm long and spaced 1.6cm apart in a plane. Nichrome electrodes were selected because of our previous experience with them in resistive wire wave gauges. The wires were held by a small frame made of 6mm diameter stainless steel tubing. The two outer wires make up one electrode, while the inner one makes up the other. The wires are electrically isolated from the frame with Teflon fittings.

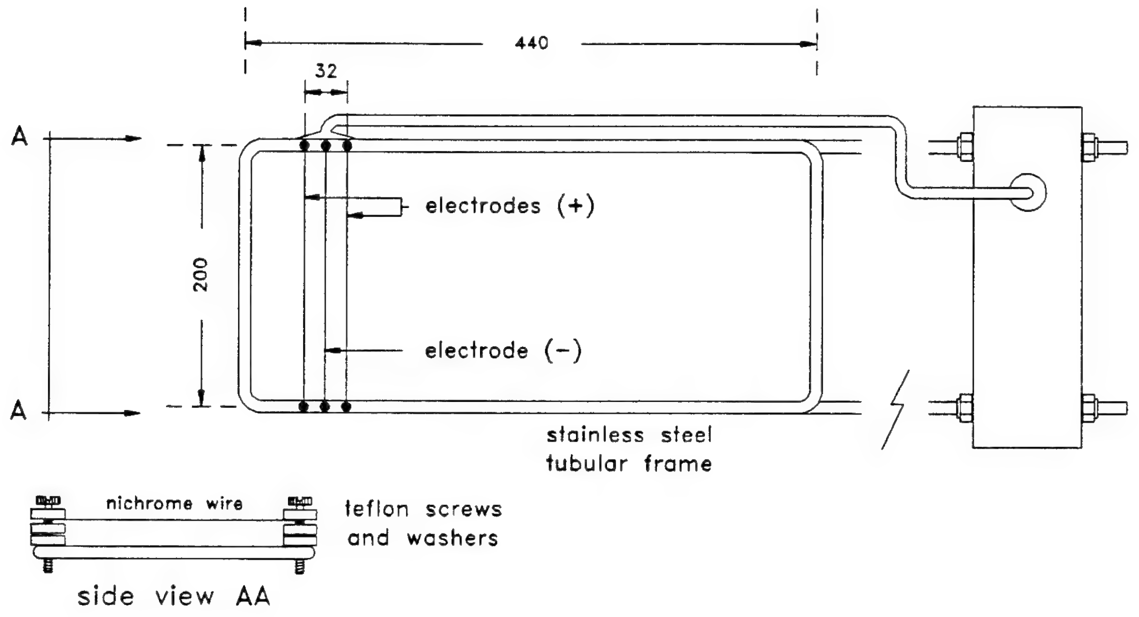


Figure 2.1: Void-fraction probe used in the laboratory experiments.

The electronics used to measure the change in electrical impedance of the bubbly mixture was a Model 80-74a AC bridge which is fabricated by the Danish Hydraulic Institute for use with their resistive wire wave gauges. The excitation frequency was 3kHz and the response of the circuitry was modified to be approximately flat out to 100Hz (3dB point). The specifications of the units are given in appendix G.

2.3 Calibration

Calibration of the void-fraction probe was performed in a cylindrical bubble tank with adjustable bubble size and air-flow rate (figure 2.2). The void-fraction in the bubble tank was measured by taking differential pressure measurements with an inverted air-on-water manometer. The manometer was an MKS model 310 pressure transducer equipped with an MKS type 170M-6B amplifier. Its resolution is 0.01% of the full range (1333 N/m^2) and its bandwidth was set to 0.04Hz. At each specific air-flow rate, the manometer and the void-fraction gauge were sampled at 200Hz and a two-minute average was used to reduce the inherent fluctuations introduced by the two-phase flow. This long time-averaging of the calibration signal allowed us to assume a steady and uniform distribution of void-fraction between the two points where pressure measurements were taken and therefore we obtain from fluid statics the following relationship for the void-fraction and the differential pressure

$$\alpha(\%) = \left(1 - \frac{p_2 - p_1}{\rho_w g h}\right) 100 \qquad (2.6)$$

where p_2-p_1 is the measured differential pressure between two points separated by a vertical distance h, ρ_w is the density of water and g is gravity. The specific density of air has been neglected in equation 2.6 since it is very small compared to that of water.

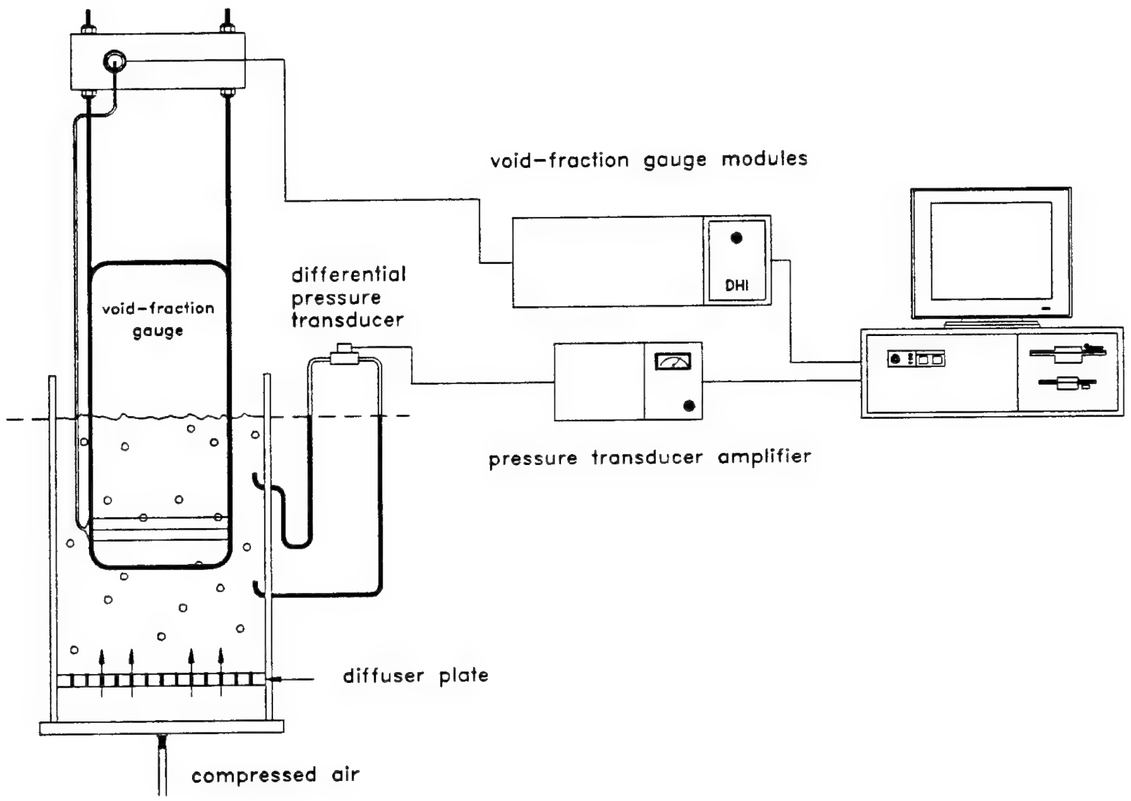

void-fraction gauge modules
DHI
void-fraction gauge
differential pressure transducer
pressure transducer amplifier
diffuser plate
compressed air

Figure 2.2: Calibration bubble tank and apparatus for the calibration of the void-fraction probe.

The probe has been calibrated for void-fractions up to 30% and data at 100% were obtained by calibrating in air. The height of the reservoir (figure 2.2) fixed the 30% upper-bound for tank calibration. Temperature changes of the water over the course of the entire calibration procedure were monitored and remained within ±0.5°C. Drift in the output of the manometer was recorded before and after each (two minutes) calibration point and was found to be negligible. Calibrations of three different laboratory void-fraction probes are shown in figure 2.3.

It should be pointed out that at low void-fraction (<10%), the bubbles are essentially uniformly distributed throughout the entire calibration tank. At higher void-fraction, we have observed the formation of a 5 to 10cm tall central bubble column located immediately above the diffuser plate with return flows on the sides of the tube. Above ~10cm, we could not observe such organization anymore and the bubbles filled the full cross-section of the tube with no visible anisotropy. The taps for the differential pressure measurement (which yielded the void-fraction) were located at 14 and 34cm above the diffuser plate and the void-fraction probe was positioned at ~24cm. Therefore we do not expect that the calibration measurements could have been biased by an anisotropic organization of the bubbles inside the calibration tank.

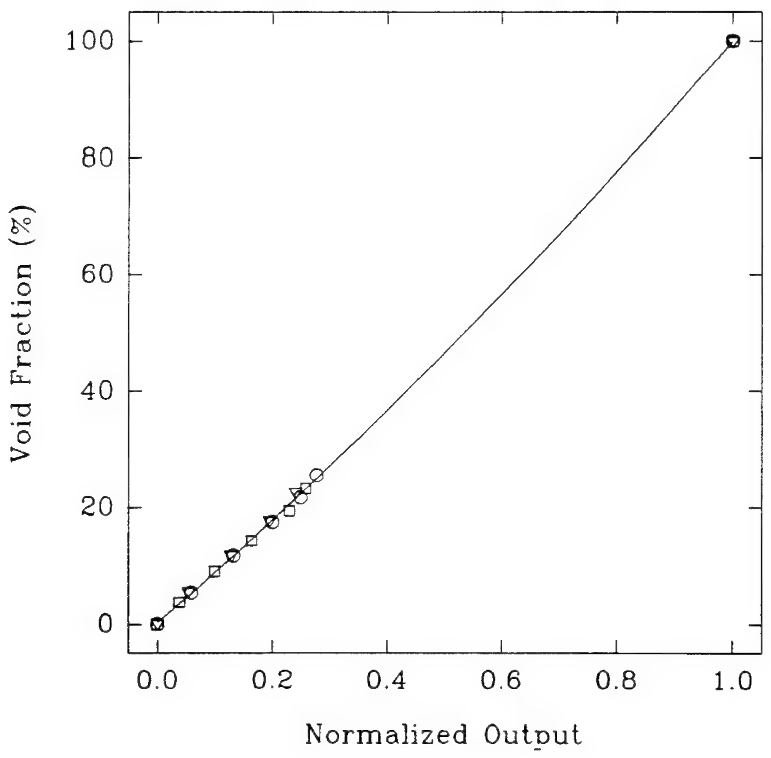


Figure 2.3: Typical calibrations of the laboratory void-fraction probes. The normalized output is given by $V^* = (V_w-V)/(V_w-V_a)$ where V is the voltage output from the instrument, V_w is the voltage in water only and V_a is the voltage in air only. Direct calibrations up to 30% void-fraction were obtained along with calibration at 100% in air. The three different symbols represent calibrations of three different probes. Data is fitted with a second-order polynomial.

The rms noise of the instrument was approximately 0.1% (void-fraction) with some electronics modules offering lower noise levels. We defined a detection threshold of 0.3% (void-fraction) to be unambiguously above the rms noise of the instrument. Signals above the detection threshold were therefore due to bubbles in the water and not to electronic noise.

2.4 Detail investigation of the void-fraction probe performance

In this section we investigate the sensitivity of the void-fraction probe to changes in water temperature and bubble size. We also look into the dimensions of the measuring volume and we define a minimum depth for which void-fraction measurements are not affected by the presence of the water surface.

2.4.1 Temperature effect

The conductivity of bubble free water is a strong function of temperature and salinity with temperature being more important in typical oceanic conditions. Generally speaking, this dependance varies on a very slow time scale (O(0.1°C/hr)) in water basins the size of a wave tank (or larger) compared to the larger and faster fluctuations in conductivity caused by air entrainment in breaking waves (order 1% void-fraction or more per second). Therefore, these slow changes due to temperature appear as DC offsets for the void-fraction measurements and can be easily subtracted from the signal. Nevertheless, breaking itself introduces small, local fluctuations in temperature which occur on a fast time scale (comparable to void-fraction). These fluctuations have not been directly measured in bubble plumes immediately after breaking but in the context on this work we can estimate them to be less than 1m°C from our knowledge of the size of the bubble plume generated by breaking and the total energy dissipated during breaking. It should be pointed out that Thorpe and Hall [1987] have used a towed thermistor array to measure the temperature anomalies due to bubble clouds at depths of 0.9m to 8.6m. They have found fluctuations up to 25m°C which are caused by inhomogeneous temperature distributions from the entrainment of warmer or colder air by breaking waves.

The dependence of the void-fraction measurement on temperature has been found by increasing the bubble free water temperature from 14°C to 32°C and by recording the void-fraction gauge DC output. The equivalent void-fraction due to the temperature increase can be computed from the calibration and it is shown on figure 2.4 where we find a bias of 0.4% void-fraction per degree Celsius. Since the void-fraction noise threshold of the laboratory probe described in this work is about 0.3% we can therefore assume temperature fluctuations due to breaking to be negligible compared to the measured void-fraction fluctuations.

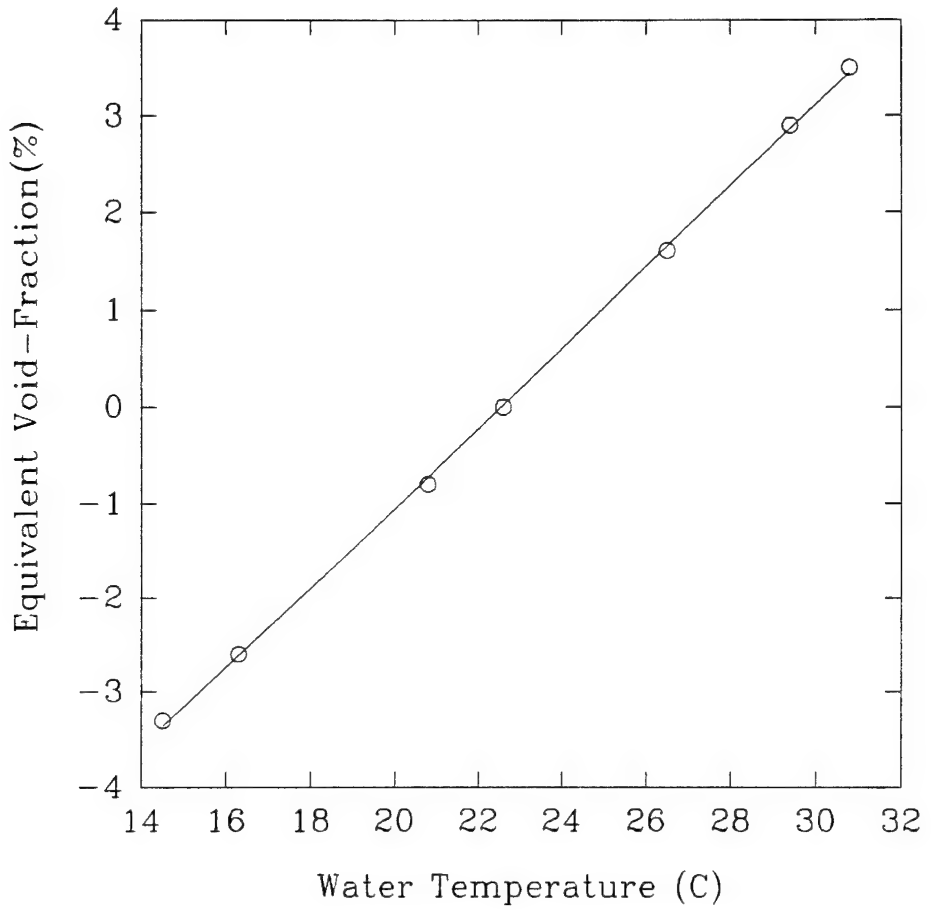


Figure 2.4: Dependence of void-fraction measurement on water temperature. The equivalent void-fraction output from the instrument is plotted as a function of the water temperature. The zero point has been arbitrarily set at 22.6°C.

2.4.2 Bubble size effect

An ideal probe would give the same void-fraction independently of the bubble size distribution. In reality, the effect of bubble size becomes negligible only if the diameter d of the bubbles is about an order of magnitude smaller than the spacing D between the electrodes [Olsen, 1967]. Bubbles of various diameters can be generated in the bubble tank by changing the hole size in the diffuser plate and their diameter can be measured from video images at low void-fraction. Figure 2.5 shows calibration of the probe for two different bubble sizes. It is clear from this plot that for a fixed void-fraction inside the calibration tank, the instrument gives a different output depending on the size of the bubbles. If we assume that calibrations with 1.5mm radius bubbles (d/D=0.19 was the smallest ratio achievable) have negligible bubble effect, we can therefore estimate the bias error for 5mm radius bubbles (d/D=0.63) to be about 10% of the void-fraction reading. The use of the statement "10% of the void-fraction reading" should be clarified. For example, if the void-fraction instrument gives a void-fraction of 30%, a possible error corresponding the 10% of this reading could be caused by bubble size effect. Hence the true void-fraction reading would fall in the range 27% to 30%. We expect 5mm radius bubbles to be representative of the largest bubble size present in bubble plumes generated by breaking waves. Indeed, video photography of breaking from the side of a wind-wave channel [Kalvoda, 1993] and subsurface video photography of bubble plumes in the field (refer to §5) have shown that the maximum bubble radius encountered was 5mm.

In our study of bubble-size effect, we have tried several other techniques to generate bubbles with radius smaller than 1.5mm. A solution of 10% by volume of iso-propyl alcohol was used to reduce the surface tension but the bubble size reduction was not significant. A fine wire mesh placed on top of the diffuser plate did not help in splitting up the bubbles. Much smaller bubbles could be generated by hydrolysis but their number density was too low to build up to a detectable void-fraction level. We have been unable to test a porous glass membrane (with nitrogen bubbles) because of the difficulty in obtaining a membrane 24cm in diameter (size of the calibration tank) and capable of supporting the pressure necessary to drive the gas bubbles through it.

In our attempt to find a way to eliminate bubble-size effect, we have experimented with a void-fraction gauge made out of two circular stainless steel electrodes 2.5cm in diameter and spaced by 10cm (we refer to this probe as the "plate" gauge). These dimensions approximately matched the dimensions of the measuring volume of the probe shown in figure 2.1. We have found the "plate" gauge to be insensitive to bubble-size effect as expected for d/D=0.1 (bubble diameter d=1cm and electrode spacing D=10cm). Unfortunately, surface effects due primarily to fringing were found to be significant for depths comparable to the electrode spacing (surface effects are discussed later). Furthermore, flow perturbation was increased by the necessity for a bulkier supporting frame for the deployment of the electrodes. In the end, the probe shown in figure 2.1 was chosen to minimize the surface effect at the expense of a possible 10% (of the void-fraction reading) bias error due to bubble-size effect.

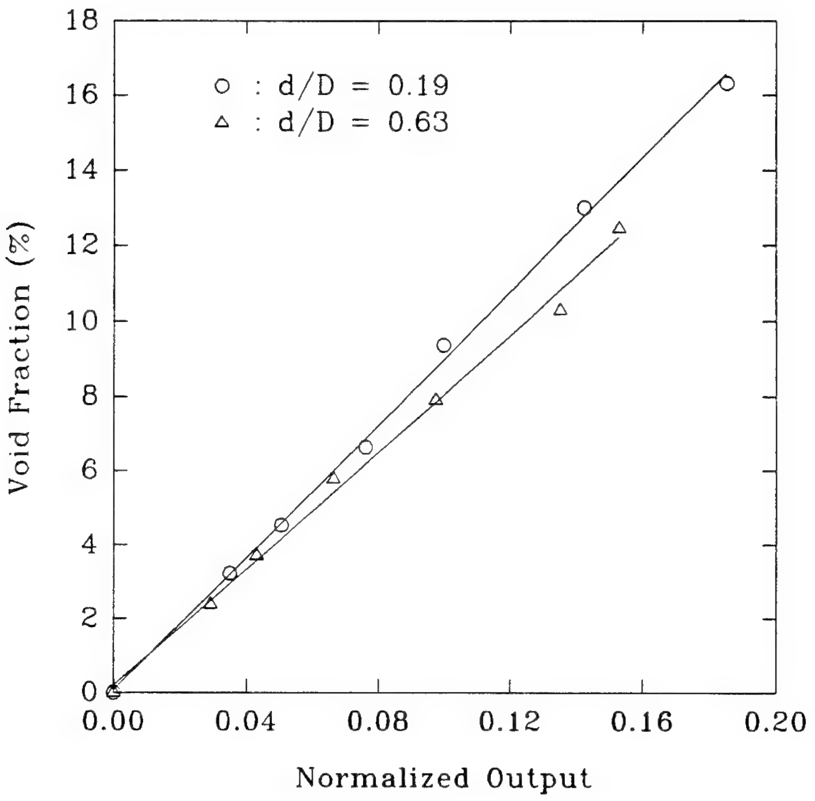


Figure 2.5: Calibration of the void-fraction probe for two different bubble sizes. The ratio of the bubble diameter to the electrode spacing is given by the parameter d/D.

2.4.3 Surface effect

A probe for free-surface void-fraction measurements must have a measuring volume small enough such that measurements close to the water surface can be made, but large enough to minimize the biasing effect of large bubbles. These two constraints are referred to as surface effect and bubble-size effect. Surface effect occurs when the electric field induced by the electrodes is modified by the proximity of the water surface. A simple test was conducted in order to quantify this effect. The void-fraction probe was initially positioned at a depth of 10cm in bubble free water. It was then raised by 0.5cm increments until the proximity of the water surface significantly changed the output.

Figure 2.6 shows the various electrode configurations tested and their dependence on the proximity of the water surface. The equivalent (bias) void-fraction output is plotted on the vertical axis. The water surface interferes with the probe's measuring volume when the equivalent void-fraction begins to rise. A comparisons with a two-wire probe with the same electrode spacing shows the advantage of the three-wire probe in applications near the free surface. The measuring volume for the probe is approximately 20cm long (i.e. the length of the wires) and of elliptic cross-section. The depths at which the surface effect become significant define the transverse dimensions of the measuring volume. The open circle symbols give the dimensions of the semi-major axis (~2.5cm based on a 0.3% threshold) and the open square symbols give the dimensions of the semi-minor axis (~1.8cm). These dimensions for the measuring volume are of the same order as the value calculated by Hill and Woods [1988] for the two-point electrode configuration with similar electrode spacing.

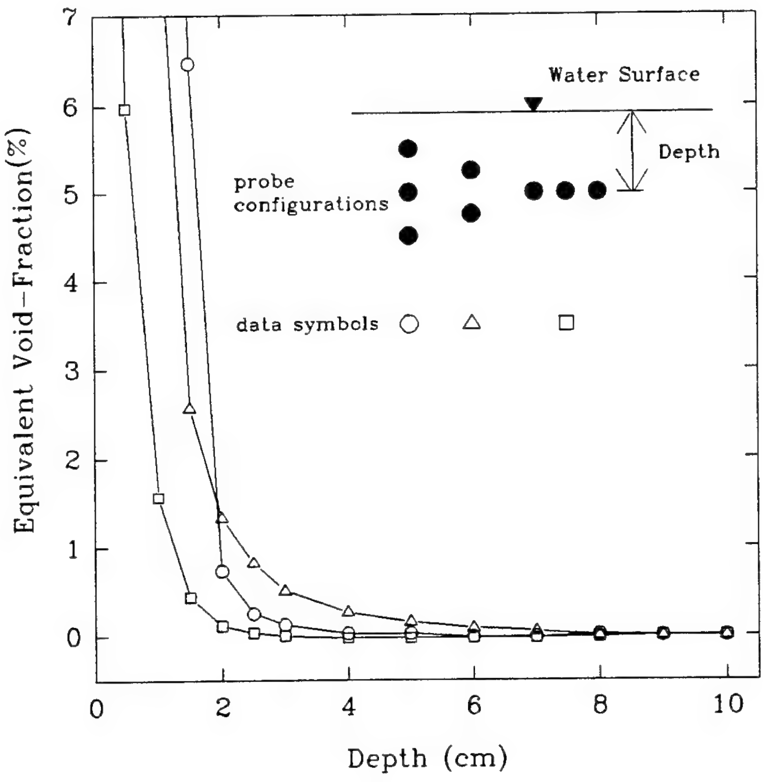


Figure 2.6: Surface effect. When the free surface is too close to the measuring volume, the void-fraction measured by the instrument is biased by the free surface. The diagram in the upper-right hand corner shows how the depth was measured for the various electrode configurations.

2.5 Field void-fraction probe

A field version of the laboratory void-fraction gauge was built for field measurements during the Surface Wave Dynamics Experiment (SWADE) [Weller et al., 1991]. The probe is decribed here and the field results are reported in §5.

The conductivity of sea water is approximately three orders of magnitude larger then the conductivity of fresh water. It is therefore clear that modifications to either the electronics or the probe are needed to optimize the void-fraction instrumentation for sea conditions. The later approach was chosen. Two 6mm diameter stainless steel electrodes 30cm long were embedded in 10mm non-conducting glass-reinforced epoxy tubes (figure 2.7). A 4cm long by 4mm wide incision was made to each of the tubes in order to expose the electrodes to the sea water. This masking was found to significantly increase the cell constant and permitted the operation in sea water with unmodified electronic circuitry. The electrodes were mounted 10cm apart on a small glass reinforced epoxy waterproof enclosure.

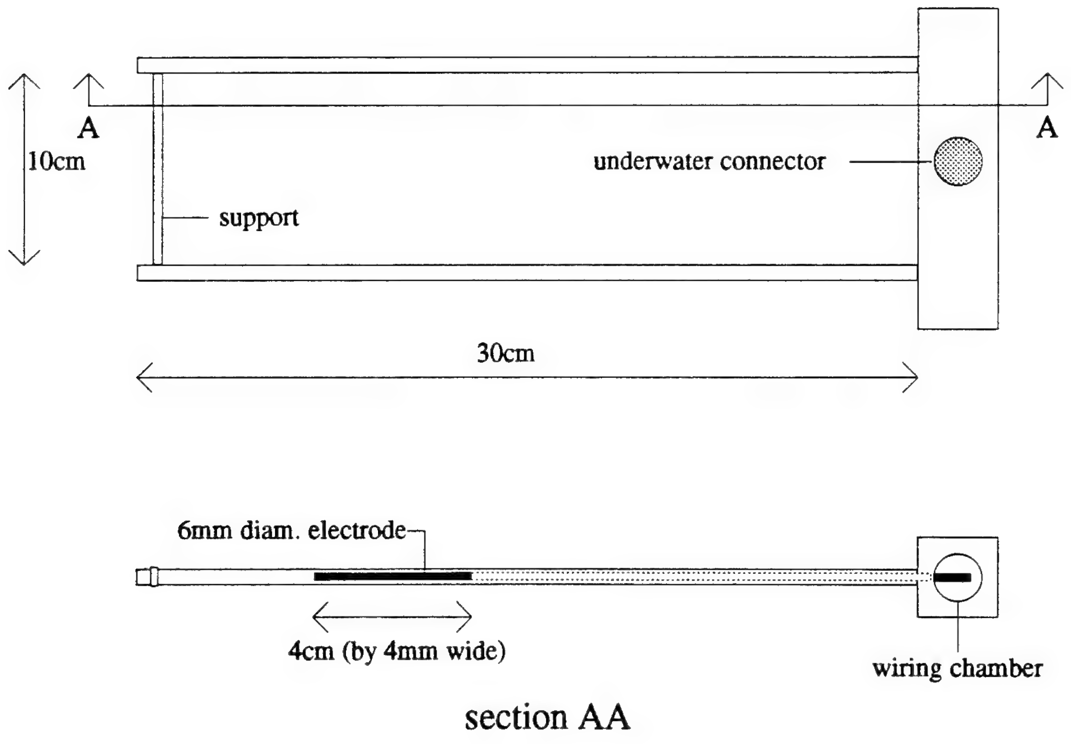


Figure 2.7: Sketch of the field probe used during the SWADE experiment.

The field probe was post-calibrated (after the experiment) in the calibration tank (figure 2.2) with sea water from the experiment site. Calibrations are shown in figure 2.8. The noise level of the field probe was found to be similar to the laboratory version (~0.1%). Signals above a detection threshold of 0.3% void-fraction were unambiguously caused by bubbles and not electronic noise. Surface effects were found to be negligible when the probe's top electrode was located at a 6cm depth. Bubble-size effects were not investigated for the field probe since the ratio of the largest bubbles encountered in the subsurface video to the electrode spacing (i.e. d/D) was about 0.1 (refer §5.4). Hence, bubble size effect for the field probe were considered negligible

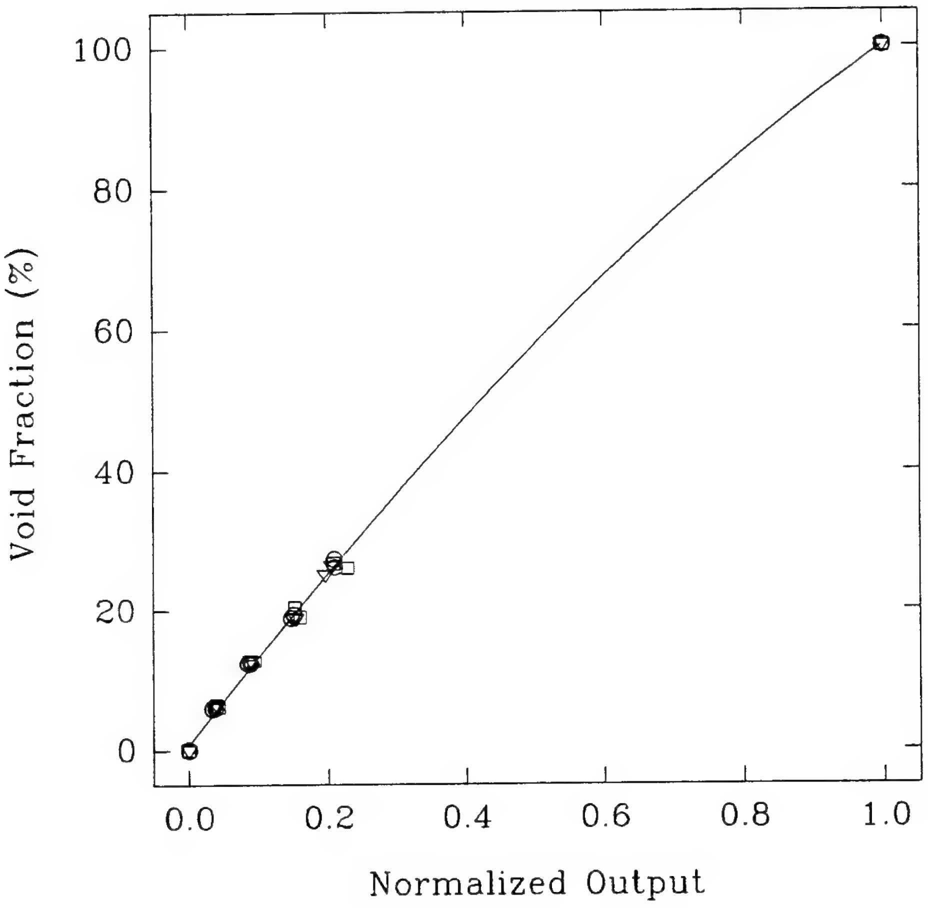


Figure 2.8: Calibrations of the field void-fraction probes. The normalized output is given by $V^* = (V_w-V)/(V_w-V_a)$ where V is the voltage output from the instrument, V_w is the voltage in water only and V_a is the voltage in air only. Direct calibrations up to 30% void-fraction were obtained along with calibration at 100% in air. The three different symbols represent calibrations of three different probes. Data is fitted with a second-order polynomial.

2.6 Discussion and summary

In this chapter, we have reported on the development of a laboratory and field instrument for the measurement of void-fraction in breaking waves and related free surface two-phase flows. The probes consisted of small electrodes of different polarity which defined a measuring volume within the bubbly mixture. The impedance of the bubbly mixture was measured with a standard commercial bridge operated at 3kHz.

The instruments were calibrated up to 30% void-fraction in a controlled bubble tank and at 100% void-fraction in air. Several important characteristics of the instrument were identified. Rapid fluctuations in the water temperature were shown to be negligible when compared to void-fraction signals above 0.3%. The probe was found capable of resolving void-fractions to within 3cm (measured from the center of the measuring volume of the probe) of the water surface before suffering from surface interference. Of course, this limitation is primarily a function of the dimensions of the measuring volume which has to be kept relatively large in comparisons with typical bubble sizes.

The instrument output signal was found to be slightly dependent on bubble size for large bubbles (i.e. 5mm in radius). For example, at the same void-fraction, a 10% difference was observed in the instrument output for bubbles with a radius of 1.5mm and bubbles with a radius of 5mm. At smaller radius, the difference is expected to vanish since the ratio of bubble diameter to electrode spacing becomes small. Video photography of bubble plumes from of a channel [Kalvoda, 1993] and subsurface video photography during the SWADE field experiment (§5.4) showed that bubbles were of radius equal to or smaller than 5mm during wave breaking events. Hence, we estimate the upper-bound error on the void-fraction measurement to be no greater than 10% (of the void-fraction reading).

Chapter 3

Void-fraction measurements in 2D laboratory breaking waves

It this chapter, we present measurements of the void-fraction field in bubble plumes generated by controlled 2D laboratory breaking waves. Some of the work reported in this chapter first appeared in Lamarre and Melville [1991,1992]. We describe in detail the experimental facility, the instrumentation and the experimental procedure. Three different breaking wave amplitudes were studied and consequently three different bubble plume sizes were investigated. Detailed space-time mapping of the void-fraction field inside these plumes is given. Moments of the void-fraction field reveal interesting information about the kinematics and the dynamics of the plume.

3.1 The experiments

3.1.1 The experimental facility

The experiments were conducted at the Ralph M. Parsons Laboratory of the Massachusetts Institute of Technology in a glassed-wall wave channel 25m long and 0.7m wide filled with fresh water to a depth of 0.6m. The wave channel is equipped with a servo-controlled, hydraulically driven piston-type wavemaker. A wooden beach with slope 1/10 extends over the last 5.5m of the channel. The wavemaker motion is controlled by an IBM XT personal computer equipped with a Metrabyte Das20 D/A card. The transfer function of the entire wavemaker system was measured and used to give the desired wave spectrum (refer to appendix C for details). Figure 3.1 shows the experimental facility and the instrumentation setup.

3.1.2 The instrumentation

The instrumentation included three void-fraction probes which were used to measure simultaneously the void-fraction at three different locations in the bubble plume. Refer to §2 for specifications. The probes were positioned on a carriage riding on rails on top of the channel walls. The carriage was equipped with a synchronous stepping motor manufactured by Superior Electric (model M061-FD02) which moved the carriage along the tank with an accuracy of ±2mm. The stepper motor was controlled by the data acquisition computer. Refer to appendix G for details on the stepper motor system.

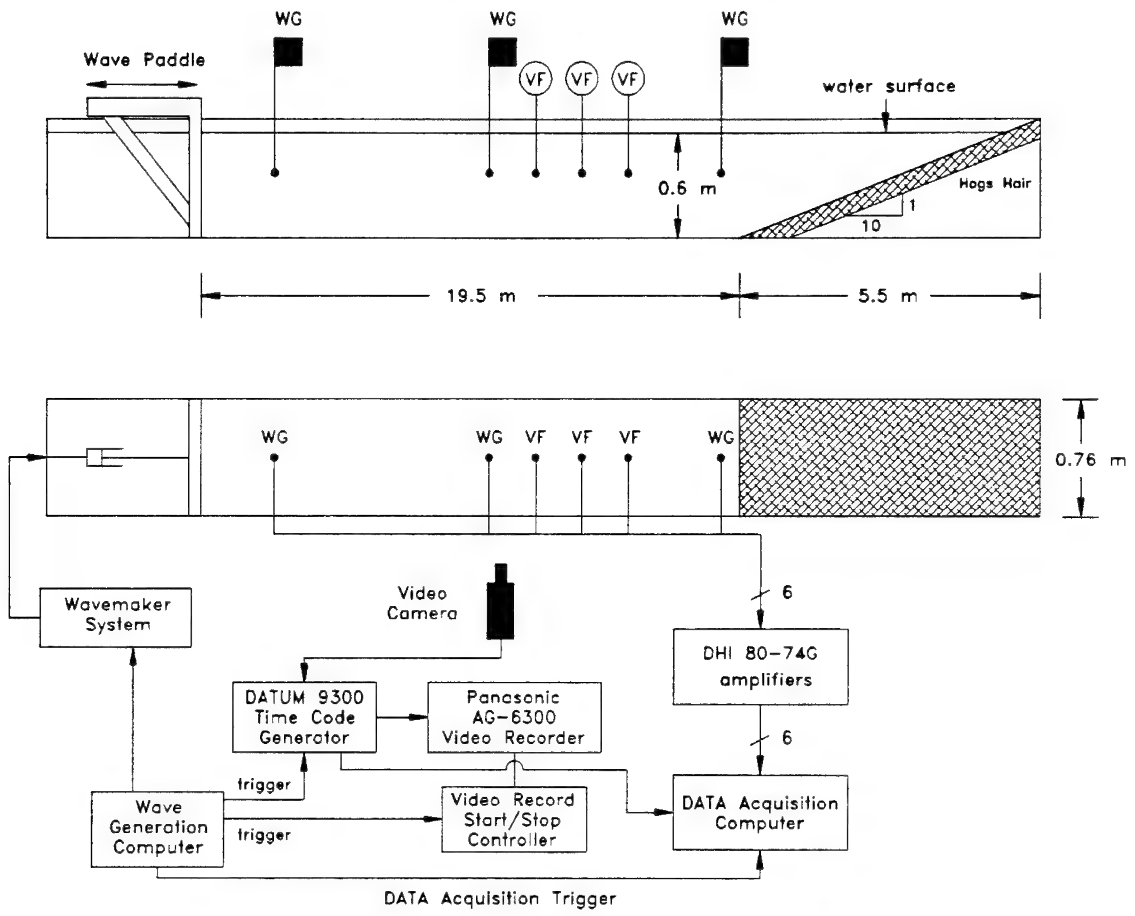


Figure 3.1: Sketch of the experimental facility and the instrumentation setup.

A set of three resistive wire wave gauges were also used to give the wave profile at several locations along the wave channel. These wave gauges were built at MIT and consist of two parallel Nichrome wires (0.127mm in diameter) 45cm long and spaced by 4mm. The electronic circuits used with these probes are commercial units from the Danish Hydraulic Institute (model 80-74G). One wave gauge was positioned on the same carriage as the void-fraction probes. It was used to determine the wave height in the vicinity of the bubble plume. The other two wave gauges were positionned upstream and downstream of the breaking location.

An NEC TI-23 1/2in format CCD video camera with 1/1000s shutter speed was used to produce videos of the breaking waves. The camera was equipped with Fujinon lenses model CF12.5A and CF25B. The video images were run through a Datum 9300A time code generator before being recorded on a Panasonic AG-6300 VHS video recorder. A triggering circuit was built to interface the recorder to the wave generation computer. The details of the circuit are given in appendix G. The video recorder was only activated when a breaking wave was being generated.

A Haaselblad 500EL/M camera was used for still photography. The lighting technique was similar to the one used by Rapp [1986]. The camera was also triggered by the computer.

3.1.3 The generation of breaking waves

The breaking waves were produced with a piston-type wavemaker by generating a packet of waves with progressively decreasing frequencies. The dispersive properties of water waves focus the packet at a predetermined position down the channel. The wave packet used in the laboratory experiment was composed of 32 sinusoidal components of equal slope a_ik_i where a_i and k_i are the amplitude and wavenumber of each component. The wave packet generation technique is discussed at length in Rapp and Melville [1990] and it is reviewed in appendix C. Various other details such as the wavemaker transfer function, the wave tank settling time and the calculation of dissipation are also given in appendix C.

Typical wave-gauge time-series of the wave packet as it propagates down the channel are shown in figure 3.2. The spectra of each time-series is also given. The measurements were taken at 3.0m, 7.7m and 15.5m from the wave paddle and breaking occurs at 8.05m from the paddle. The three wave gauge time-series show clearly how dispersion focuses the wave packet as it approaches the breaking point. After breaking, continued dispersion defocuses the wave packet. The different spectral slope at higher frequencies in figure 3.2b is due to short waves being locally generated by breaking [Rapp and Melville, 1990]. These high frequency waves quickly attenuate and do not reach the upstream and downstream wave gauges.

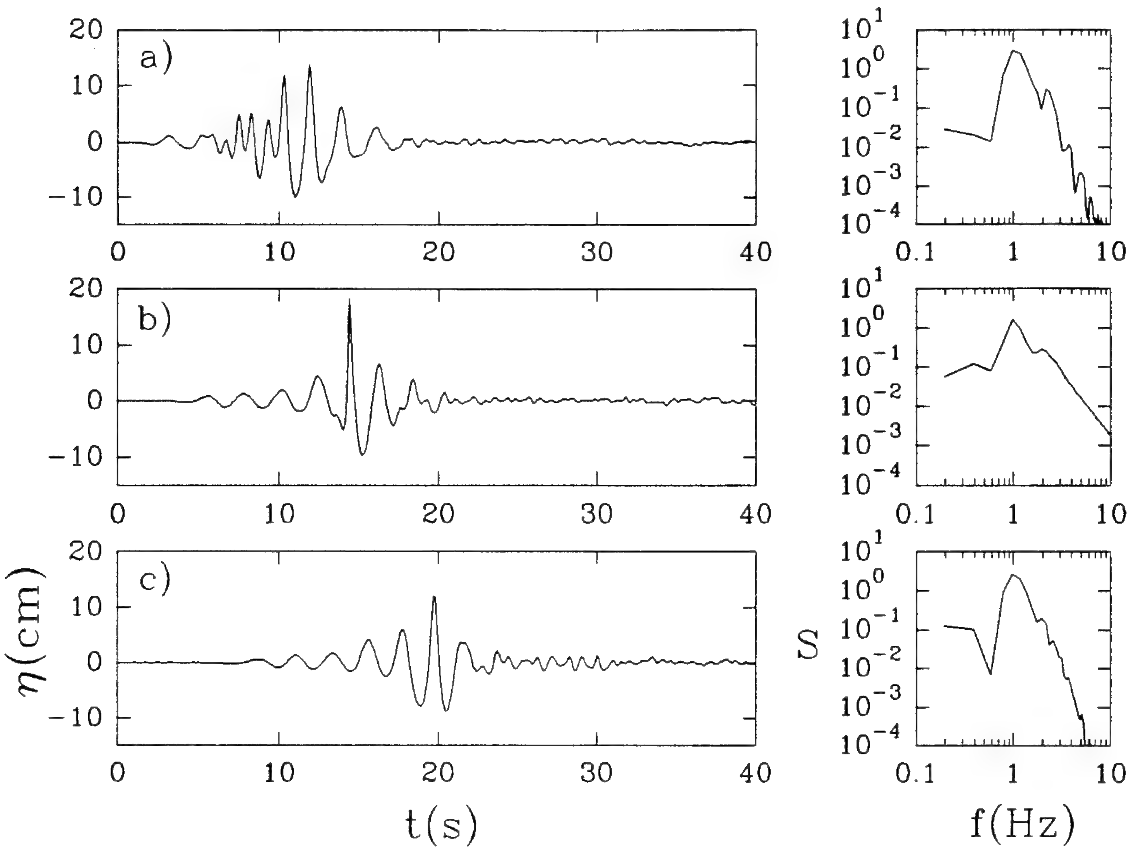
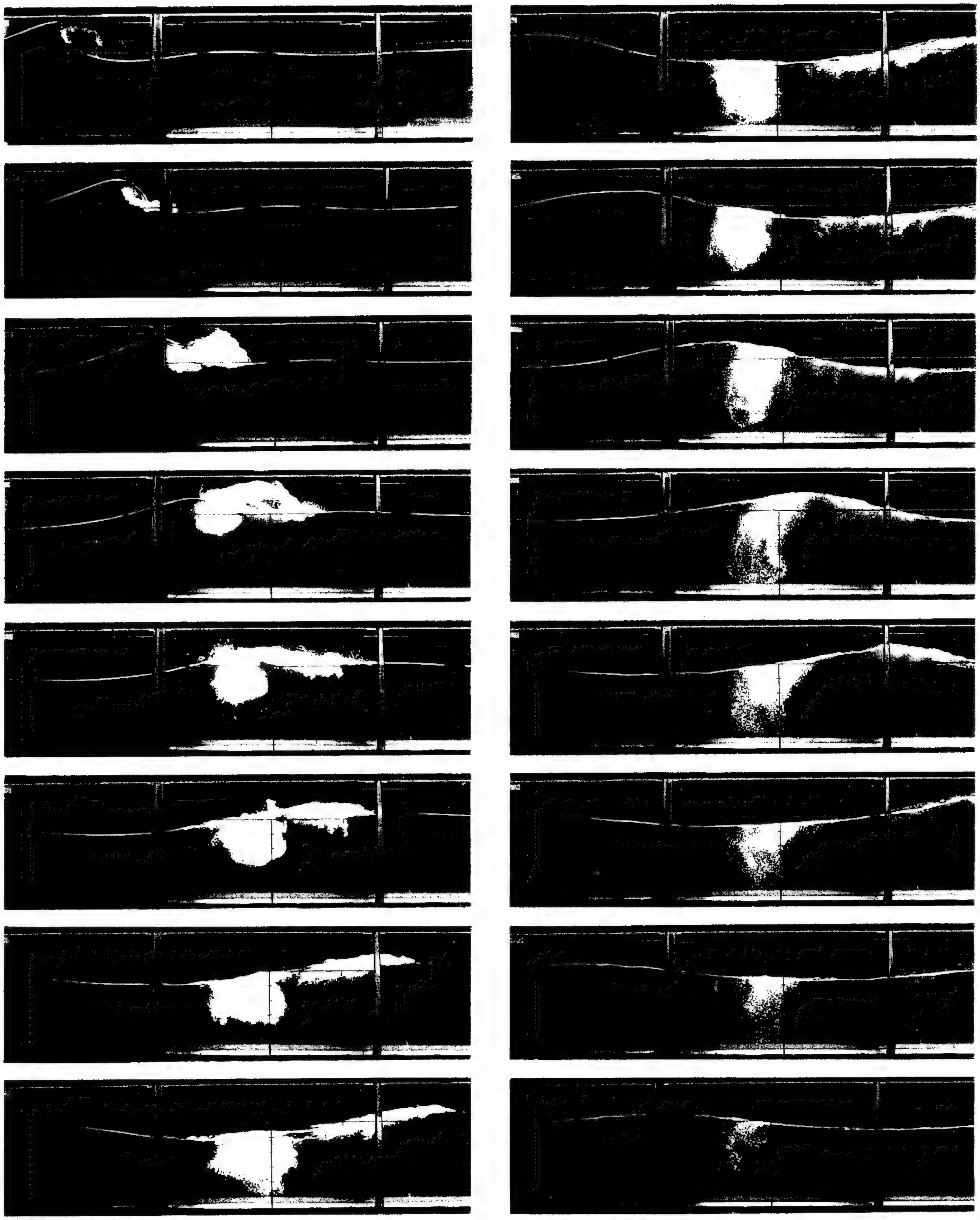


Figure 3.2: Wave gauge time-series at a) 3.0m, b) 7.7m and d) 15.5m from the paddle. The smaller plots on the right are the corresponding spectra. Breaking occurs at 8.05m. Time is referenced to initial motion of the wavemaker. Breaking occurs at t=14.4s. This data is for the wave packet S=0.54.

A series of synchronized still photographs of the wave profile from the side of the channel are shown in figure 3.3. The wave shown on these photographs was used as a 'trial wave' for performing the initial test of the void-fraction measurement technique. This wave had a slightly larger amplitude and characteristic wavelength than the waves studied in this chapter (which are described in table 3.1). Nevertheless, the shape was very similar and the photographs give a good visualization of the formation and evolution of the bubble plume.

Figure 3.3: Photographs of a breaking wave. The frames progress in time from top to bottom and from left to right. Frame 1 to 9 are separated by 0.2s and frame 9 to 16 are separated by 0.3s. The wave moves from left to right. Each photograph corresponds to a different realization of the breaking wave.



The repeatability of the wave profile was verified by making wave gauge measurements 5cm upstream of breaking for 10 repeats of the same breaking wave. The wave channel was allowed to settle for 6min between each repetition. It was found that after 6min the wave channel had reached a very calm state which did not improve with longer settling time (refer to appendix C). Results are shown in figure 3.4 where all 10 repeats are plotted. The repeatability of the wave profile is excellent. The larger differences observed at later times are due to short waves, generated by breaking, which are moving upstream towards the wave gauge.

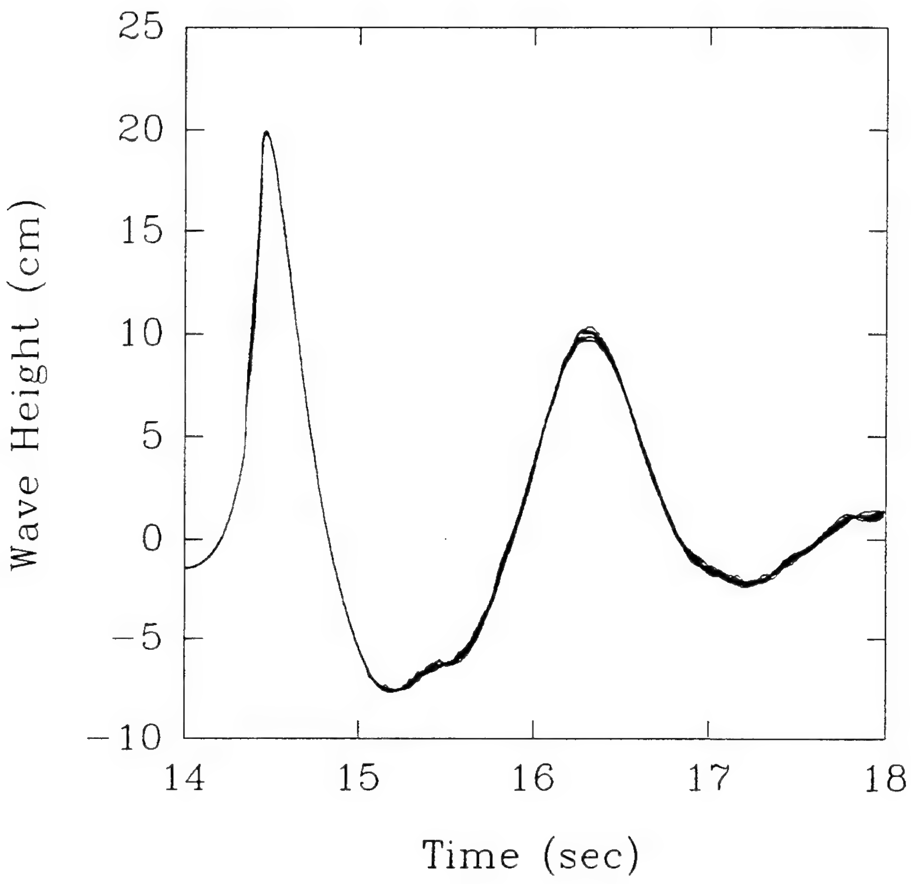


Figure 3.4: Ten wave gauge time-series 5cm upstream of breaking for 10 realizations of the same breaking wave experiment. The larger differences at later times are due to short waves, generated by the breaking wave, which are moving upstream towards the wave gauge. Time is referenced to the initial motion of the wavemaker. Breaking occurs at t=14.4s. This data is for the wave packet S=0.54.

3.1.4 Experimental procedure

The void-fraction in the bubble plume was mapped by following a grid pattern. The grid spacing was Δx by Δz where Δx is the spacing along the tank and Δz is the vertical spacing (refer to table 3.1). The probes were positioned at different locations on this grid and three repeats of the breaking wave experiment were ensemble averaged to give the void-fraction history at a specific point inside the bubble plume. This was repeated until the entire bubble plume was covered. Approximately 500 repetitions of the same breaking wave were necessary in order to completely map the void-fraction field in the bubble plume. The probes were positioned inside the channel with the long axis of their measuring volume spanning the width of the tank.

S	t_b	x_b	H	V_o	E_d	$\frac{E_d}{\rho g V_o \lambda}$	Δx	Δz	Δx_m	Δz_m
	(s)	(m)	(m)	(cm^2)	(J/m)		(cm)	(cm)	(cm)	(cm)
0.54	14.4	8.05	0.335	98.0	17.8	0.096	5	5	1	1
0.45	14.3	7.70	0.276	46.0	8.6	0.098	5	3	1	1
0.38	14.3	7.70	0.214	25.0	4.3	0.091	5	2	1	1

Table 3.1: Characteristics of the three wave-packet amplitudes studied in the 2D experiments. The wave packet was composed of 32 sinusoidal components of equal slope $a_i k_i$ where a_i and k_i are the amplitude and wavenumber of the i^th component. The slope of the wave-packet is defined as $S=\Sigma a_i k_i$. The center frequency of the wave packet and the bandwidth were both 0.88Hz. The characteristic wavelength of the wave packet is λ. The time from initial motion of the paddles up to breaking is t_b which is taken as the time when the forward-moving jet of the breaker strikes the free surface at a distance x_b from the paddles. The maximum measured wave height at $x=x_b$ is H. V_o is the total volume of air per unit width enclosed by the forward jet as it impacts the free surface and it was measured from video images taken from the side of the channel. E_d is the total energy dissipated by breaking per unit width. The measurements of the void-fraction were taken on a grid of size Δx by Δz and interpolated on a grid of size Δx_m by Δz_m.

Three breaking waves were studied. They were all generated from the same wave packet but in each case, a different gain was applied to the signal. This generated three similar shape breaking waves with three different amplitudes. Table 3.1 shows the characteristics of the three wave packet amplitudes studied. We will often refer to the largest amplitude wave packet as the most energetic breaker since it dissipated the most energy (refer to E_d in table 3.1). It should be pointed out that E_d is the energy per unit width dissipated by breaking only. It was corrected for viscous dissipation caused by the bottom and the walls of the channel (refer to appendix C).

3.2 The void-fraction measurements

Figure 3.5 shows typical time-series of the wave gauge and two void-fraction gauges. All sensors were located at the same horizontal position along the channel. The two void-fraction time-series are for probes located at -5cm and -15cm below the still water level. The dashed line in the wave gauge time-series indicates the position of the uppermost void-fraction probe at -5cm.

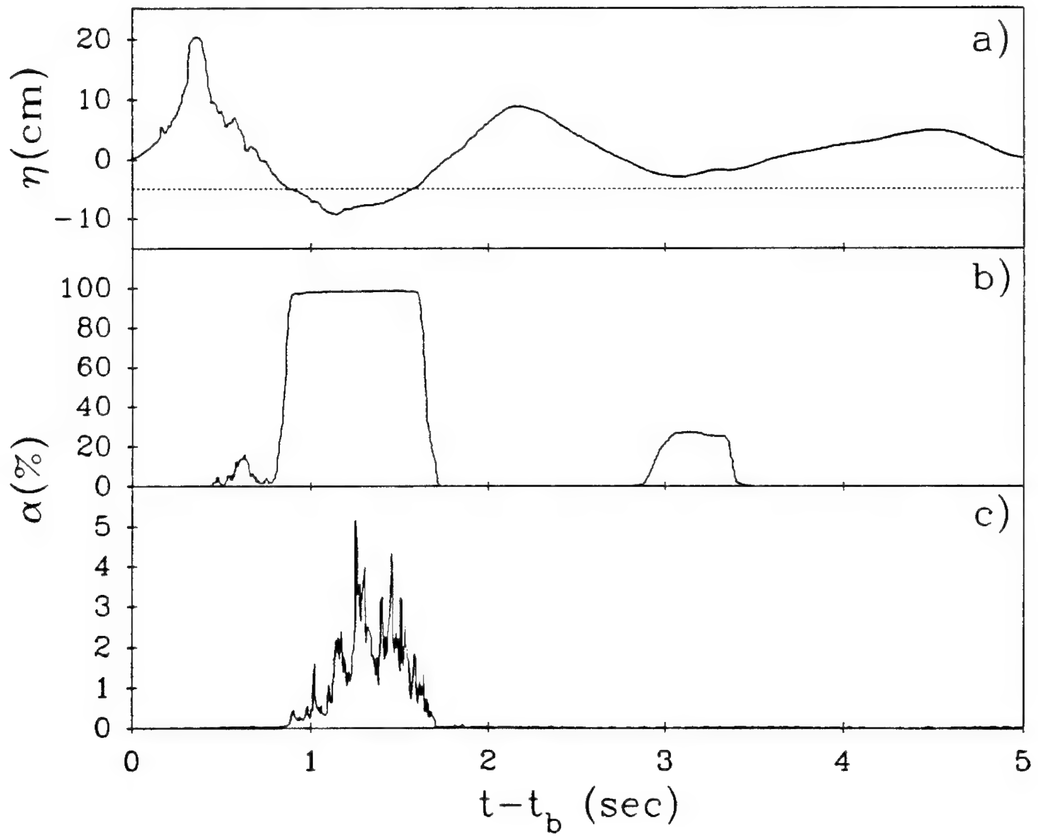


Figure 3.5: Typical time series at 200Hz of a) wave gauge measurements b),c) void-fraction measurements located at depth -5cm and -15cm from the still water level. All data are for the most energetic breaker studied at 0.55m downstream of x_b. Note how the probe crosses the water surface at 0.8sec in b). 'Surface effect' is shown in b) at 3sec. The second wave trough has its water surface at approximately -2.5cm and therefore intrudes into the measuring volume of the void-fraction gauge which is located at -5cm ±2.5cm.

The crossing of the dashed line by the wave profile figure 3.5 means that the void-fraction probe is coming out of the water. When this occur, large excursions in the void-fraction signal are generated. Also notice that for the second wave trough, the surface does not quite cross the dashed line but the signal on the void-fraction probe still shows an effect caused by the surface. We refer to this as 'surface effect'. It is due to the wave surface intersecting the measuring volume. Since the probe has a measuring volume of 5cm diameter, it will therefore extend by 2.5cm above and below its marked position (in this case -5cm). This limitation restricts reliable measurements to depths greater than 2.5cm. Signals generated by the surface were edited out, including the short segments when the signal rises and falls. Video imaging of the void-fraction probes from the side of the channel helped in this task.

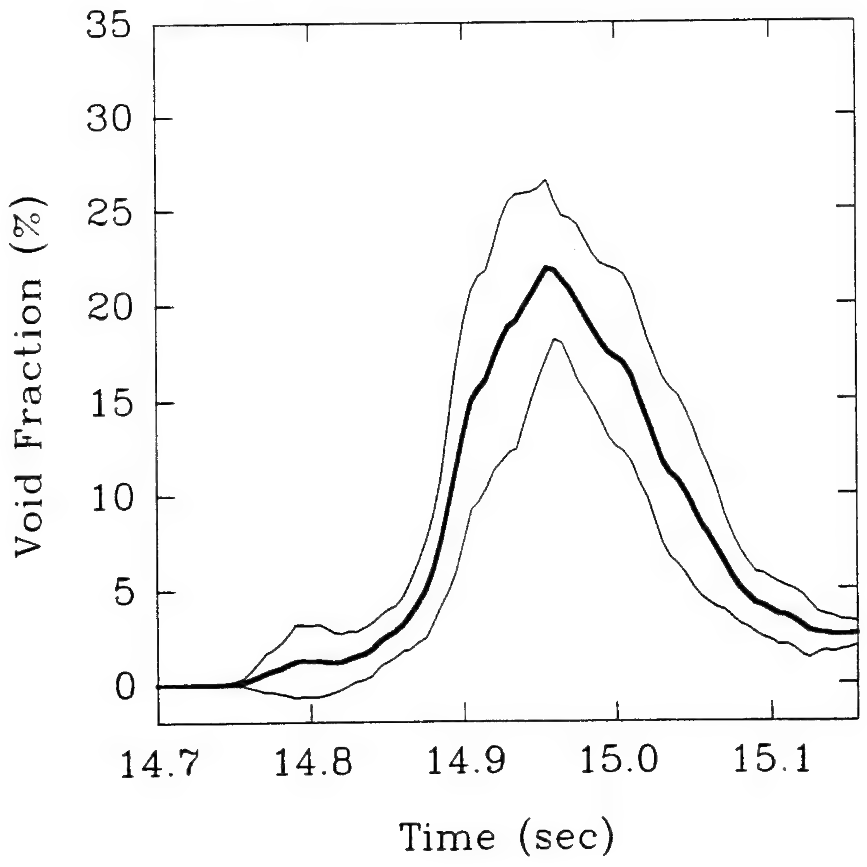


Figure 3.6: Typical repeatability of the void-fraction measurements. The mean of 20 ensemble averages of three repeats (bold line) is plotted along with ± one standard deviation of the 20 ensembles.

At each location of the sensors inside the bubble plume, three void-fraction time-series from three repeats of the experiment were ensemble averaged to reduce the inherent fluctuations generated by the turbulent two-phase flows. Figure 3.6 shows the typical repeatability of the void-fraction measurements when ensemble averaged. Sixty repeats of the experiment were divided in 20 ensemble averages of three runs. For these experiments, the void-fraction probe was positioned in the center of the bubble plume where the void-fraction levels are high. The mean and the standard deviation for this group of 20 ensemble averages are shown in figure 3.6. Overall, the repeatability of the ensemble averages is relatively good.

Once the entire bubble plume has been surveyed, the void-fraction field inside the bubble plume can be reconstructed from the ensemble-averaged time series. This is only possible because the measurements are repeatable and all repeats share a common time basis. The data is first edited to eliminate the transitions caused by 'surface effects'. The individual runs are then ensemble averaged and the result is time averaged over a 0.05s window. Each ensemble averaged time-series gives information about the void-fraction history at a particular position inside the bubble plume. One can therefore reconstruct, with that information, the void-fraction field inside the bubble plume at any time during its evolution.

Figure 3.7 shows color contour maps of the void-fraction field and still photographs of the wave profile and bubble plumes at four different times. The void-fraction measurements for those color maps were taken on a grid of dimensions Δx=10cm by Δz=5cm. The color maps were obtained by interpolating the original data over a finer mesh of size Δx_m=2cm by Δz_m=1cm. A Laplacian and spline scheme was used to interpolate the data [Young and Van Woert, 1989]. The inherent assumption behind this intorpolation process is that the void-fraction inside the bubble plume varies smoothly. The color maps show good agreement with the bubble plume cross-sectional area on the photographs. Void-fractions below 0.3% were below the detection threshold of the instrument and could not be resolved.

We have observed that bubbles located in the close vicinity (~5cm or less) of the vertical walls tend to get trapped, with their velocities significantly reduced in this boundary layer. This effect is more apparent when the bubbles rise back to the surface since it causes the bubbles close to the wall to reside longer. Photographs from the side of the channels (last photo of figure 3.7) show an apparently larger bubble plume (especially at later times) than what is actually present and measured by the void-fraction probe at the center of the channel.

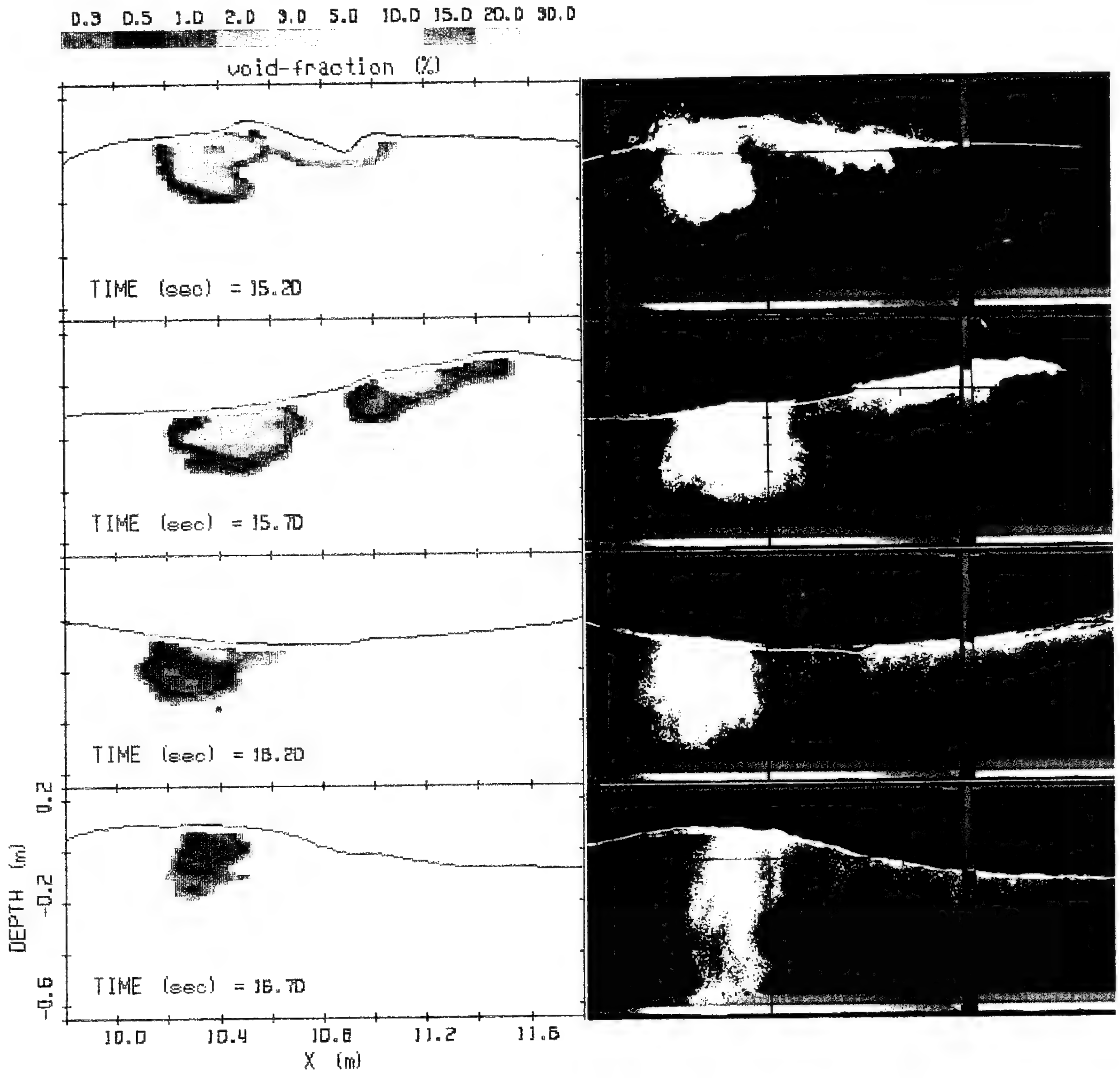


Figure 3.7: Color contour maps of the void-fraction field at four different times along with matching photographs of the breaking wave. The center line of the photographs is at 10.5m from the wavemaker paddle. The still water level is the horizontal line. Time is from the start of the paddle motion.

The data for the color maps show in figure 3.7 was from a preliminary experiment intended to demonstrate the feasibility of the technique in mapping the void-fraction field. The success of these initial tests led us to make a more comprehensive set of experiments where the dependence of air entrainment parameters on the dynamical wave parameters were studied.

Three breaking wave packet amplitudes were studied. The specific characteristics of the packet are given in table 3.1. Color contour maps similar to the ones shown in figure 3.7 were generated. Figure 3.8 shows such a map for the wave packet S=0.54 which was the most energetic breaker studied. The data show that measurements over the full 0-100% range have been observed and void-fraction in excess of 20% for all three wave amplitudes studied were sustained for up to half a wave period after breaking.

3.3 Moments of the void-fraction field

Various moments of the void-fraction field can be computed from the color maps of figure 3.8. These moments are shown in figure 3.9 and figure 3.10. The total entrained air volume per unit width shown in figure 3.9a is computed from

$$V = \int_A \alpha \, dA \qquad (3.1)$$

where α is the local void-fraction and A is the cross-sectional area of the bubble plume. It should be pointed out that V is the volume per unit width.[1] The data shown in figure 3.9a was normalized by V_o, the initial volume of air enclosed by the wave crest as it collapses onto itself. Figure 3.3 shows clearly the formation of this air pocket. This initial volume of air was measured by digitizing video images taken from the side of the channel. The three different data symbols represent the three different wave amplitudes studied. The data collapse well on an decaying exponential which suggests that V_o was the appropriate choice for normalizing the volume of air entrained.

[1] This notation is different from the one used in Lamarre and Melville [1991]. The volume V and the potential energy of the bubble plume E_b (to be discussed shortly) are expressed per unit width. This notation will facilitate comparisons with the results of §4 on the measurement of void-fraction in 3D breaking waves.

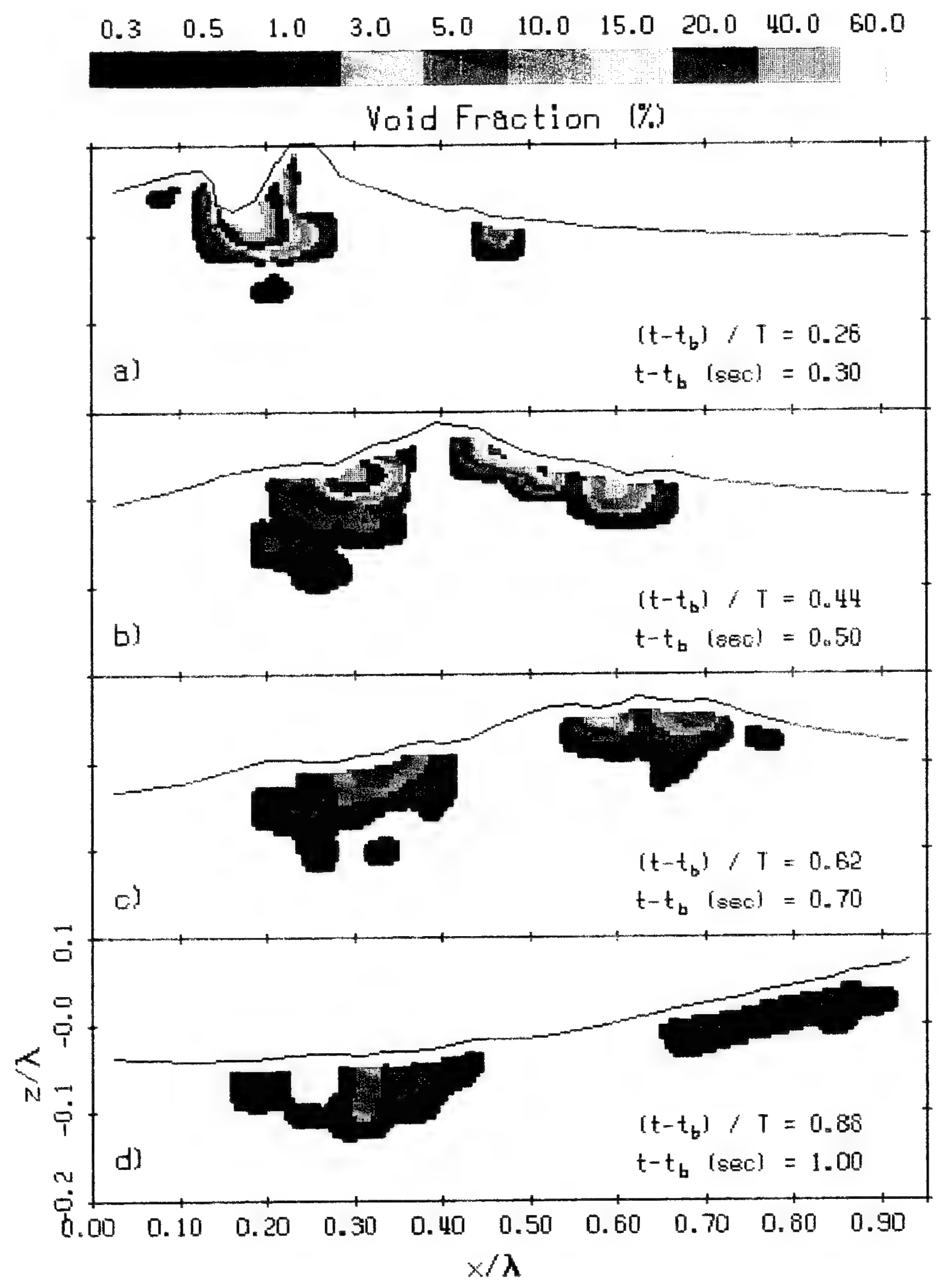


Figure 3.8: Contour maps of the void-fraction field for the most energetic breaker studied. Time evolves from a) through d). The wave moves from left to right. Each frame is constructed from approximately 170 ensemble averages of three runs. Note the distinct downstream cloud generated by the splash from the initial impact at breaking.

The measurements show that the volume of air enclosed in the initial air pocket is conserved for up to 1/4 wave period after breaking. The degassing of the bubble plume follows an exponential law and is very rapid with only 5% of the initially entrained air remaining after a full wave period after breaking. The fast rising speed of the larger bubbles (radius>1mm) accounts for most of the air lost in the first wave period. The equation for the exponential fit is

$$\frac{V}{V_o} = 2.6\, e^{-3.9\, t^*}$$

where $t^* = (t - t_b)/T$ and t_b is the time when the forward-moving jet strikes the free surface and T=1.1s is the characteristic wave period of the wave packet.

The bubble plume cross-sectional area is shown in figure 3.9b. The cross-sectional area is computed from

$$A = \int_A dA. \tag{3.2}$$

The data were normalized by V_o and fitted by using the functional form for V divided by that for $\bar{\alpha}$, the average void-fraction (to be discussed next). This procedure was used because extrapolation of the curves for V and $\bar{\alpha}$ is evidently better conditioned than extrapolation of the data for A directly. The fitted curve has equation

$$\frac{A}{V_o} = 325\, t^{*\,2.3}\, e^{3.9\, t^*}.$$

The cross-sectional area of the plume expands rapidly up to 0.5T after which the area decreases more slowly. The initial data suggest that the cross-sectional area of the plume expands at a rate of $28V_o/T$ for the time between 0.1 to 0.3T. It is anticipated that the measured cross-sectional area will be somewhat dependent on the ability of the probe to measure small void-fraction levels. Recall that the void-fraction probe detection threshold is 0.3%. The initial rise in bubble plume area is expected to be sensibly independent of the void-fraction detection threshold since during that stage, the plume is very compact and the void-fraction levels are high. During the later stage (after 0.5T), the void-fractions have diminished significantly and it is likely that the probe does not detect the small void-fraction levels on the perimeter of the plume. A detailed analysis of the effect of a finite detection threshold on the estimates of the cross-sectional area (as well as other moments) is given in §4.

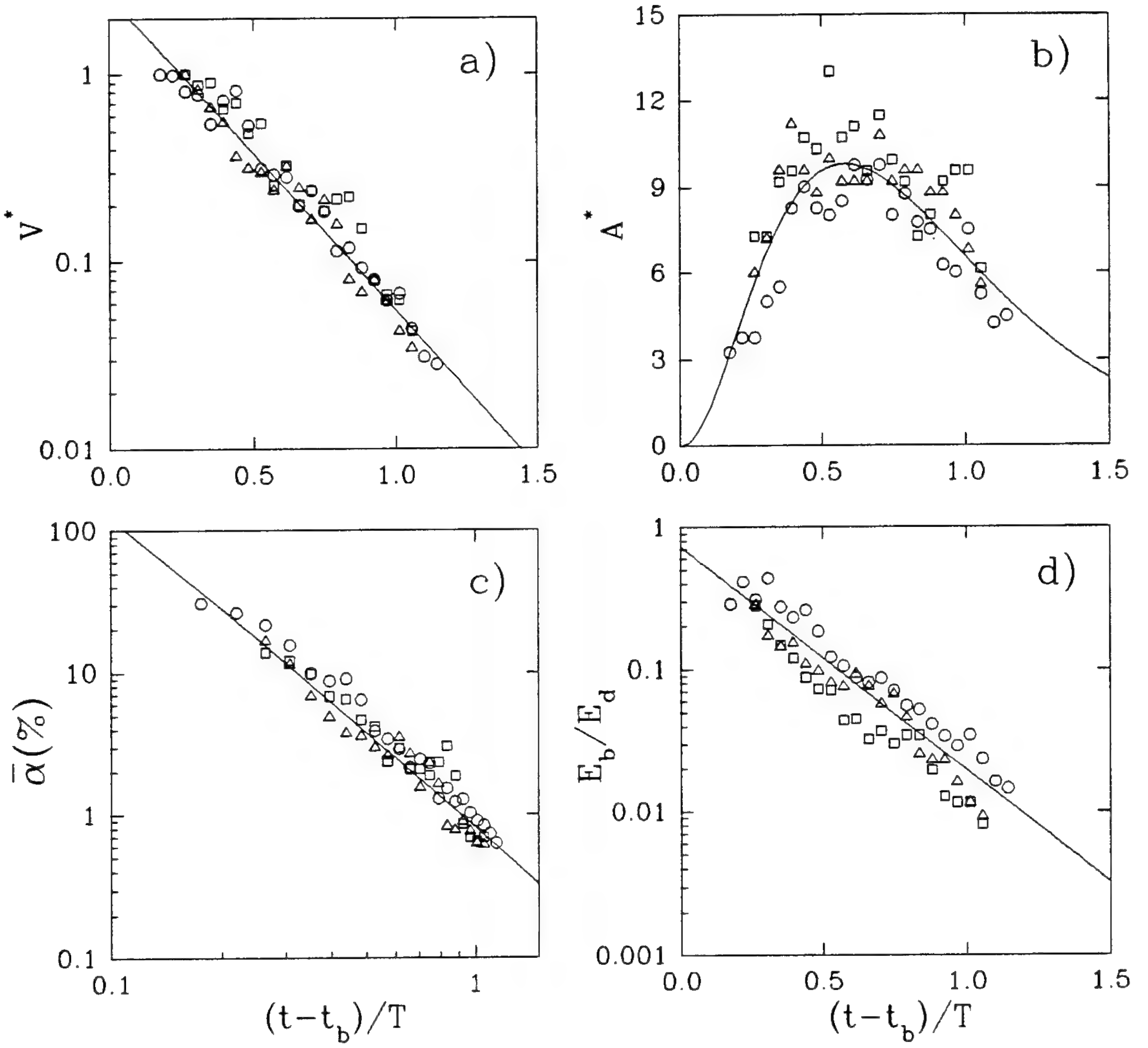


Figure 3.9: Moments of the void-fraction field for the 2D experiments. a) Normalized volume of air entrained $V^* = V/V_o$, b) normalized bubble plume cross-sectional area $A^* = A/V_o$, c) mean void-fraction $\bar{\alpha}$, d) potential energy E_b of the plume per unit width, normalized by E_d, the energy dissipated by breaking per unit width. Solid lines are best exponential fits for V^* and E_b/E_d , and best power law fit for $\bar{\alpha}$. The data for A^* has been fitted by using the functional form for V^* divided by that for $\bar{\alpha}$. The values for V_o and E_d used in normalizing V, A and E_b are given in table 3.1. Symbols are S=0.54 (O), S=0.45 (□) and S=0.38 (Δ) where S is defined in table 3.1.

The mean void-fraction is given by

$$\bar{\alpha} = \frac{V}{A}. \tag{3.3}$$

The mean bubble plume void-fractions shown in figure 3.9c remain above 1% within the first wave period after breaking. The data collapses well onto a power law

$$\bar{\alpha}(\%) = 0.8\, t^{*\,-2.3}\ .$$

Extrapolation of the data to $t \sim t_b$ suggest that the mean void-fraction in the air pocket formed at breaking is close to 100% which is consistent with the photographs taken from the side of the channel and shown in figure 3.3.

The work per unit width required to keep the air entrained against the buoyancy force is

$$E_b = \rho g \int_A \alpha z \, dA \tag{3.4}$$

where ρ is the water density, g is gravity and z is the vertical distance to the free surface. Normalizing the data by the total wave energy dissipated by breaking gives a simple exponential form as shown in figure 3.9d. The energy dissipated by breaking is calculated by taking the difference in surface displacement variances between wave gauge measurements upstream and downstream of breaking (refer to appendix C). The data are fitted with a decaying exponential

$$\frac{E_b}{E_d} = 0.7\, e^{-3.6\, t^*}\ .$$

Melville and Rapp [1990] have shown that up to 40% of the total pre-breaking wave energy can be lost through breaking. The data presented here now shows that a large fraction (30% to 50% and maybe more) of the energy lost may be expended in entraining the bubble plume.

The horizontal and vertical centroids of the bubble plume void-fraction field are respectively

$$\bar{x} = \frac{\int_A \alpha x \, dA}{\int_A \alpha \, dA} \qquad (3.5)$$

$$\bar{z} = \frac{\int_A \alpha z \, dA}{\int_A \alpha \, dA} \qquad (3.6)$$

where x is the horizontal distance from x_b and z is the depth from the free surface. Notice that the vertical centroid $\bar{z}$ describes the vertical position of the plume with repect to the free surface or equivalently its effective penetration depth below the water surface. Hence, the effect of the orbital velocity of the wave which will cause the plume to move up and down with each wave crest and wave trough is not included in this moment. However, the amount of vertical excursion caused by the wave can easily be estimated from the wave height measurement at breaking given in table 3.1.

The horizontal centroid shown in figure 3.10a moves at approximately the phase speed of the wave $C = \lambda/T$ for the initial T/2 period. The data were normalized by the characterisitc wavelength of the wave packet λ=1.94m. Notice that the centroid is computed for the entire void-fraction field. It includes both the primary and secondary bubble plume. From figure 3.8, it is clear that the formation of the secondary bubble plume tends to shift the horizontal centroid forward. Furthermore, the propagation speed of the secondary plume is faster because it is riding on the forward face of the wave. Hence, the propagation speed of the primary plume is less than the phase speed of the wave and that of the secondary plume is slightly greater. In fact, when considered independently of the secondary plume, the primary plume moves at approximately 0.7C.

The data for the vertical centroid are shown in figure 3.10b where they have been normalized by H the wave height at breaking (refer to table 3.1). The data for the most energetic breaking wave (S=0.54) show, after some initial scatter, that the vertical centroid deepens with time at a speed 0.2H/T and reaches a maximum depth of ~0.40H. The trend for the data with S=0.45 remains unexplained.

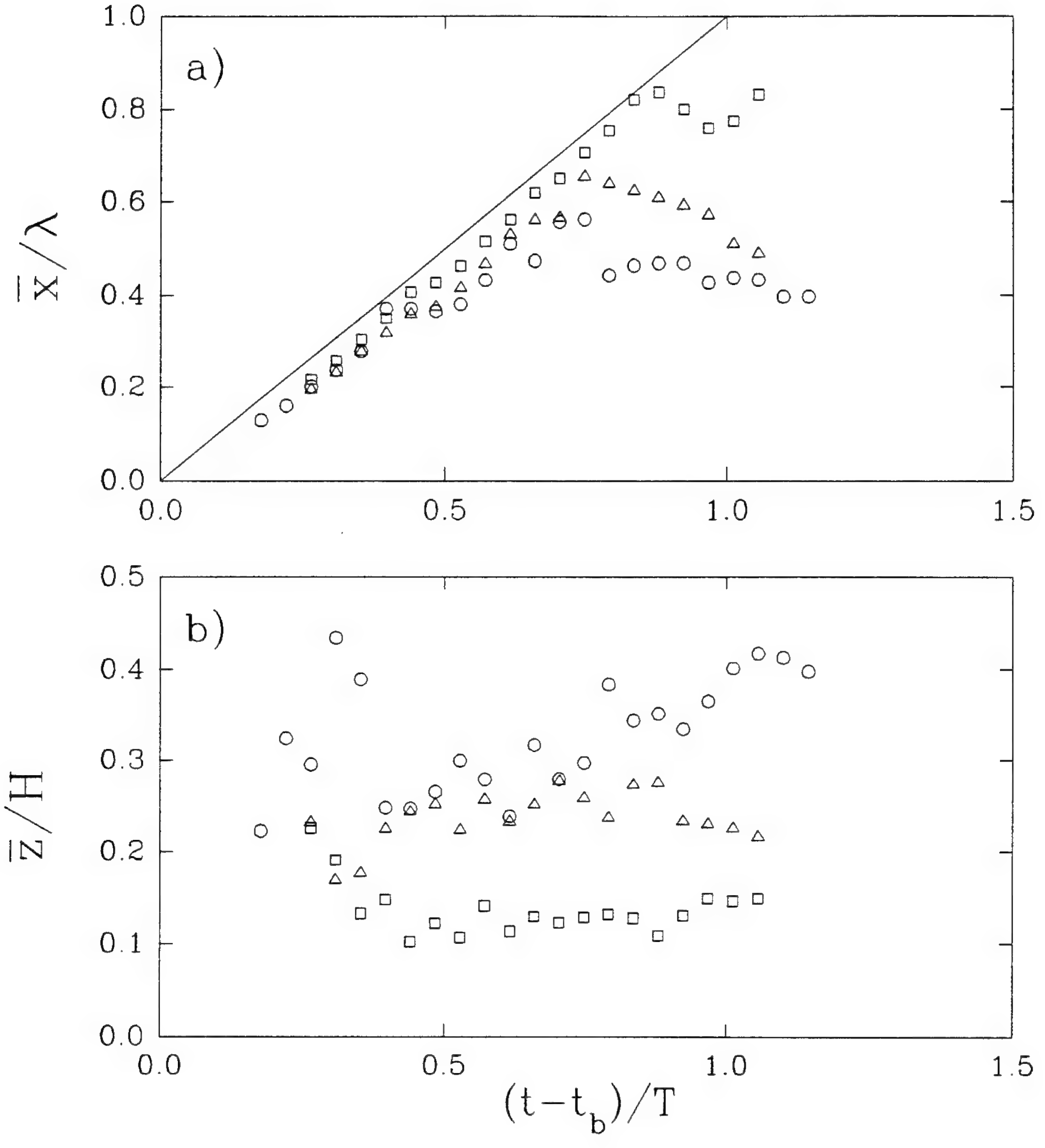


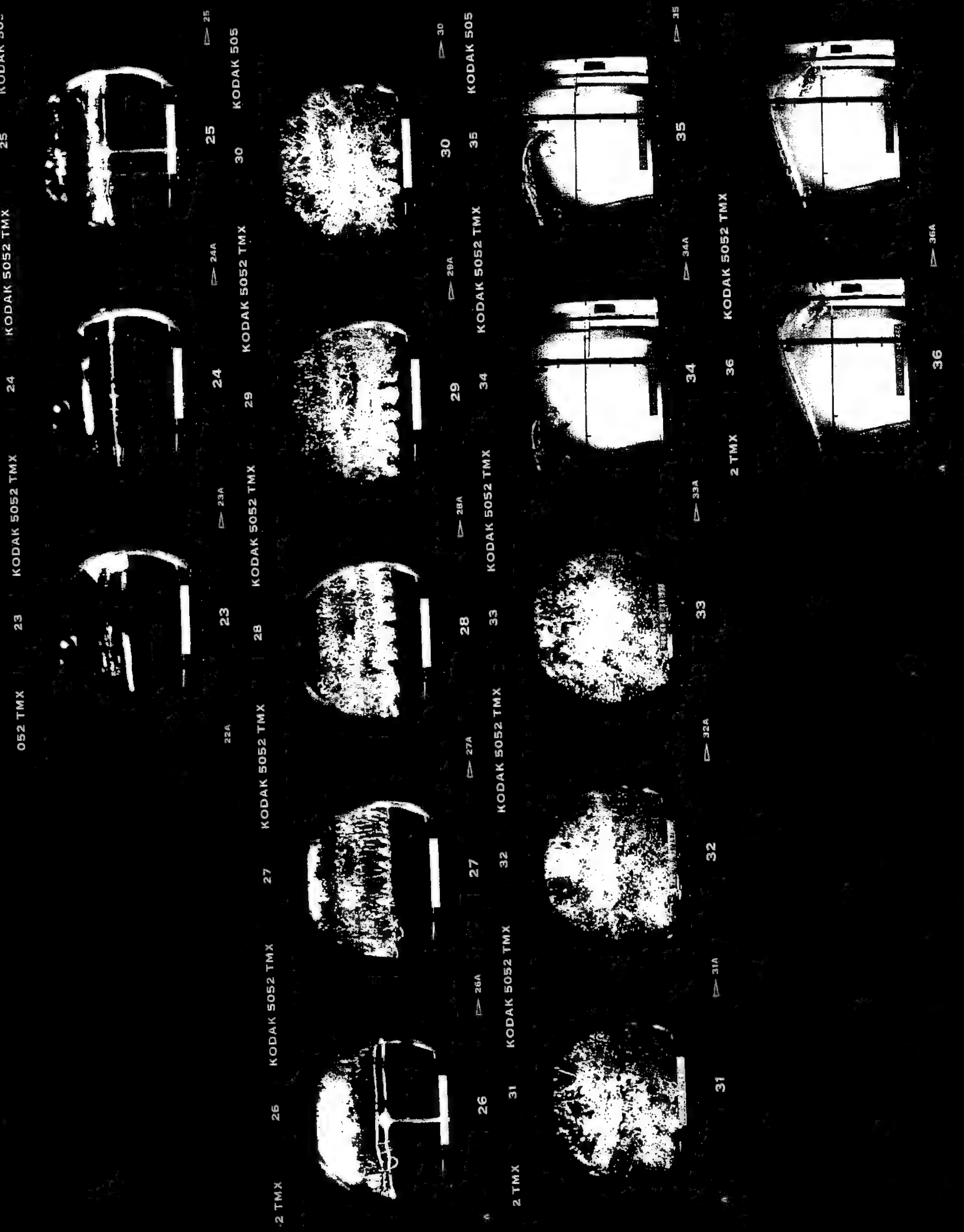
Figure 3.10: Centroids of the bubble plume. a) Horizontal centroid $\bar{x}$ of the bubble plume normalized by λ=1.94m, the characteristic wavelength of the wave packet. b) Vertical centroid $\bar{z}$ normalized by H the height of the wave at breaking (see table 3.1). Solid line is the phase speed of the wave C = λ/T. Symbols are S=0.54 (O), S=0.45 (□) and S=0.38 (Δ) where S is defined in table 3.1.

3.4 Breaking of the air cylinder

A series of underwater videos were taken to study the breakup of the cylinder of air formed at the onset of breaking. An NEC TI-23A video camera equipped with a Fujinon HF4.8B-SND4-1 wide angle lens was mounted inside an underwater video enclosure by Video Vault (Spring, Texas). The video camera was positionned at approximately mid-depth inside the wave channel. The camera was looking horizontally towards the wave paddle and thus the breaking wave progressed in the direction of the camera. Videos from the side of the channel were also taken. Photographs of the video monitor with the recorder set on still mode were taken and are shown in figure 3.11 [see Loewen, 1991 for details on the photographic technique used].

The caption of figure 3.11 describes in detail the evolution and breakup of the air cylinder. Perhaps the most interesting features are the instabilities observed on the wall on the air cylinder prior to breaking up (frame 27) and the finger-like bubbly structures (frame 29) which seem to match the original spacing of the instabilities. The instabilities are probably caused by a combination of centrifugal and gravitational effects. The transition from perfectly smooth air cylinder (frame 26) to completely turbulent bubbly mixture (frame 31) takes only 0.4s in this example. In this time period the breakup process must go from one large volume to an ensemble of small air bubbles of various sizes. The evolution of the bubble population during that period is important to quantify since it will largely determine the amount of gas transfer realized during that stage.

Figure 3.11: (On the following page). A series of photographs showing the breakup of the air cylinder. The camera is located inside the wave channel and looking toward the wave paddle for frames 23 to 33 and the breaking wave is coming direclty towards the camera. The time down to milliseconds is indicated on each frame. Frame 34 to 36 are from the side of the channel. In frame 23, the cylinder of air is about to be formed. The lowest horizontal line is the interface between air and water. This particular frame corresponds to frame 35 from the side of the channel. In frame 25, the forward face of the wave impacts the water surface and bubbles start to appear at the impact zone. This corresponds to frame 36 from the side of the channel. In frame 26, we observe a smooth cylinder of air with bubbles at the surface. In frame 27, the cylinder has penetrated the surface even more and instabilities start to appear. In frame 28 and 29, breakup of the cylinder is initiated and small finger-like structures with spacing comparable to the instabilities of frame 27 are formed. Frame 31 and up show the breakup evolving into a complex turbulent bubbly mixture. Video by Eric Lamarre, photos of the video by Mark Loewen [1991]



KODAK 5052 TMX
23
24
25
26
27
28
29
30
31
32
33
34
35
36

3.5 Discussion and summary

In this chapter, we have reported on a series of experiments on 2D laboratory breaking waves. The evolution of the void-fraction field inside the bubble plumes generated by breaking was successfully measured for three different breaking wave amplitudes (figure 3.7 and figure 3.8). The measurements showed that the volume fractions of air near the surface within one wave period following breaking were many orders of magnitude greater than the time-averaged values previously measured in the field [Walsh and Mulhearn, 1987].

Several moments of the void-fraction field have been computed (figure 3.9 and figure 3.10). They show that the bulk properties of the bubble plume generated by breaking waves scale well with prebreaking wave variables such as the initial enclosed air volume at breaking V_o and the total energy dissipated by breaking E_d. Furthermore, these moments were found to evolve according to simple functions of time.

The degassing of the plume was found to be very rapid with only 5% of the initially entrained air remaining in the water a full wave period after onset of breaking. Laboratory wind-flume experiments [Broecker and Siems, 1984] have shown that the contribution of bubbles to the total air-sea flux of carbon dioxide by bubbles is almost independent of radius in the range 0.02 to 2mm despite the fact that larger bubbles are less numerous and have shorter underwater residence time. Therefore large bubbles which have a lifetime of a wave period or less can contribute significantly to the transfer of carbon dioxide. The very fast degassing rate of the bubble plume revealed by the present study indicates that modelling of the contribution of bubbles to total air-sea gas transfer may be considerably underestimated if the transient large bubble population (radius ~0.1 to 10mm) is not considered.

The mean void-fraction in the bubble plume remained above 1% within the first wave period after onset of breaking. This has important consequences in the field of underwater acoustics. It is well known that the speed of sound in water can be reduced by an order of magnitude in the presence of void-fraction as low as 1% (refer to appendix A). Hence, significant sound-speed changes are associated with these large localized void-fraction fields. These sound-speed discontinuities have been speculated to exist and regarded as possible sources of low frequency sound (below 1kHz) in the ocean. Prosperetti [1988] and Carey and Browning [1988] have proposed that bubble plumes could emit low frequency sound through coherent volume oscillations of the entire plume which would be similar to the oscillation of an air bubble in water. Loewen and Melville [1993] have performed acoustic measurements of the sound generated by the breaking waves studied in this chapter. The void-fraction measurements reported in the present study permitted prediction of the resonance frequency of the plume according to a model developed by Lu et al. [1990]. The predicted and measured frequencies matched very well giving strong support to the hypothesis that low frequency sound is generated by volume oscillations of the bubble plume. The results of these comparisons will be shown in §4.8 along with the data for 3D breaking waves.

It was shown that a significant fraction (30 to 50%) of the energy dissipated by breaking is used in entraining the bubbles against their buoyancy. This is direct evidence that surface wave evolution and air entrainment are dynamically coupled. Furthermore, the approximately constant ratio $E_d/\rho g V_o \lambda$ for these experiments as shown in table 3.1 implies that the initial volume of air entrained correlates with the energy dissipated. This was first pointed out by Lamarre and Melville [1991] and later confirmed by Loewen and Melville [1993] who conducted measurements of V_o and E_d on a broad range of laboratory breaking waves. More specifically, they varied the amplitude and the characteristic wavelength of the packet to obtain the data shown in figure 3.12. The solid data points are from the present study and the hollow data points are from Loewen and Melville [1993]. The volume V_o is plotted versus the dissipation due to breaking $D_b=E_d/E_w$ where E_w is the total energy contained in the wave field prior to breaking. The results show a clear correlation between the total volume of air entrained and the energy dissipated.

Recent studies have shown that the wave energy dissipated correlates with the acoustic energy radiated and the microwave radar backscattered by breaking waves [Loewen and Melville, 1991]. Thus, the correlation between air entrainment and dissipation shown in figure 3.12 suggests that it may be possible to quantify air entrainment and gas transfer in the field using acoustic and microwave techniques.

The kinematics of the bubble plume showed that the centroid of the plume was initially advected horizontally by the phase speed of the wave. The primary plume was found to move at approximately 0.7C. Field studies [Thorpe et al., 1983], using sonar, found that the cloud in the early stage after breaking moved at the wind-wave phase speed. More recent field experiments [Ding, 1992] which have used an horizontal array of hydrophones to track the position of the breaking waves, have shown that the average phase speed of the breaking wave was between 0.45C to 0.75C where C is the phase speed at the peak of the surface displacement spectrum. The upper range of these findings are consistent with our laboratory results. Finally, for the most energetic breaking wave studied, the penetration depth of the vertical centroid was found to increase as a function of time.

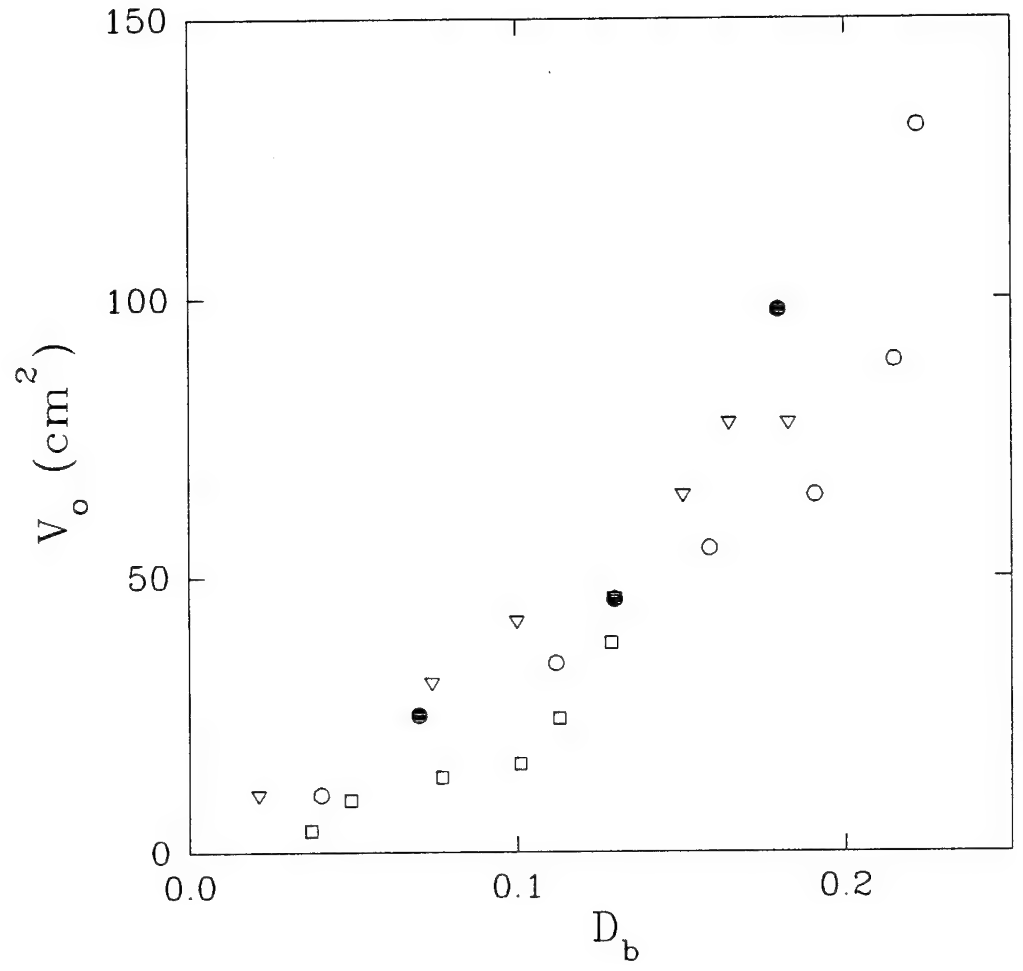


Figure 3.12: Total volume of air entrained V_o per unit width versus fractional energy dissipated by breaking $D_b = E_d/E_w$ where E_d is the energy dissipated by breaking (without visous loss on the bottom and walls of channel) and E_w is the total energy in the wave packet prior to breaking. The solid symbols are from the present study. The hollow symbols are from three different wave packets studied by Loewen and Melville [1993]. (Adapted from Loewen and Melville [1993]).

It remains to comment on the applicability of these results to the field. The experiments presented here were conducted in fresh water. There is evidence that fresh and sea water bubble densities can differ by an order of magnitude [Monahan and Wang, 1992]. However, immediately after the formation of a bubble plume the population of large bubbles in fresh water and sea water is similar and these large bubbles account for most of the entrained air. Monahan and Wang [1992] have found that the maximum void-fraction found immediately after breaking (obtained from integrating their measured bubble population) did not differ substantially between fresh and salt water but that important differences were observed at later times when the void-fraction is low. These findings suggest that the results presented in this chapter on the measurements of high void-fractions in the first wave period following breaking would also apply in sea water for these short times after breaking.

The problem of scaling laboratory results to full size ocean waves is always a difficult one. A good starting point would be to compare the results of this chapter with those of §4 (this is done at length in §4) which were obtained with 3D laboratory breaking waves. The 3D waves were comparable to small breaking waves at sea with maximum wave heights of 0.8m and wavelengths of 6m (these parameters are three times greater than those of the 2D experiment). In general, the moments of the void-fraction field in the 2D and 3D experiments scaled well with the wave height and the wavelength suggesting that scaling to field conditions may be applicable.

Chapter 4

Void-fraction measurements in 3D laboratory breaking waves

In this chapter, we report on detailed measurements of the evolution of the void-fraction field in bubble plumes generated by large-scale three-dimensional (3D) laboratory breaking waves. Some of the work reported in this chapter first appeared in Lamarre and Melville [1993]. The experimental facilities and experimental procedure are presented first. We follow with detailed measurements of the space-time evolution of the void-fraction field. Various moments of the void-fraction field are calculated and compared with results from the two-dimensional (2D) experiments (refer to §3). The radial dependence of the void-fraction and sound-speed field inside the bubble plumes is also investigated. Finally, we point out that the frequency of the collective oscillations of the bubble plume predicted from the void-fraction measurements compare favorably with measurements of the low frequency pressure fluctuations [Loewen and Melville, 1993]. Hence, strong support is given to the hypothesis [Prosperetti, 1988; Carey and Browning, 1988] that low frequency sound is generated by the collective oscillations of the bubble plume.

4.1 The experiments

4.1.1 The facilities

The experiments on 3D breaking waves were conducted in the large multi-directional wave basin facility at the Offshore Technology Research Center (OTRC) of Texas A&M University in June, 1991. The basin is 45.7m long and 30.5m wide and filled with fresh water to a depth of 5.8m (figure 4.1). One end of the basin is outfitted with several wave absorber metal screens deployed over the full depth. The other end is equipped with 48 hydraulic wave paddles hinged at a depth of 3m and controlled by a DEC workstation for the signal generation, and three 80386 PC computers for the D/A conversion. A motorized platform spanning the basin was used to mount the instrumentation. The breaking waves were obtained by radially focusing (see radial crest lines in figure 4.1) a wave packet made up of progressively decreasing frequencies. The center frequency of the wave packet and the bandwidth were approximately 0.5Hz and

1.0Hz respectively. The dispersive properties of water waves focused the packet at a predetermined position down the basin.

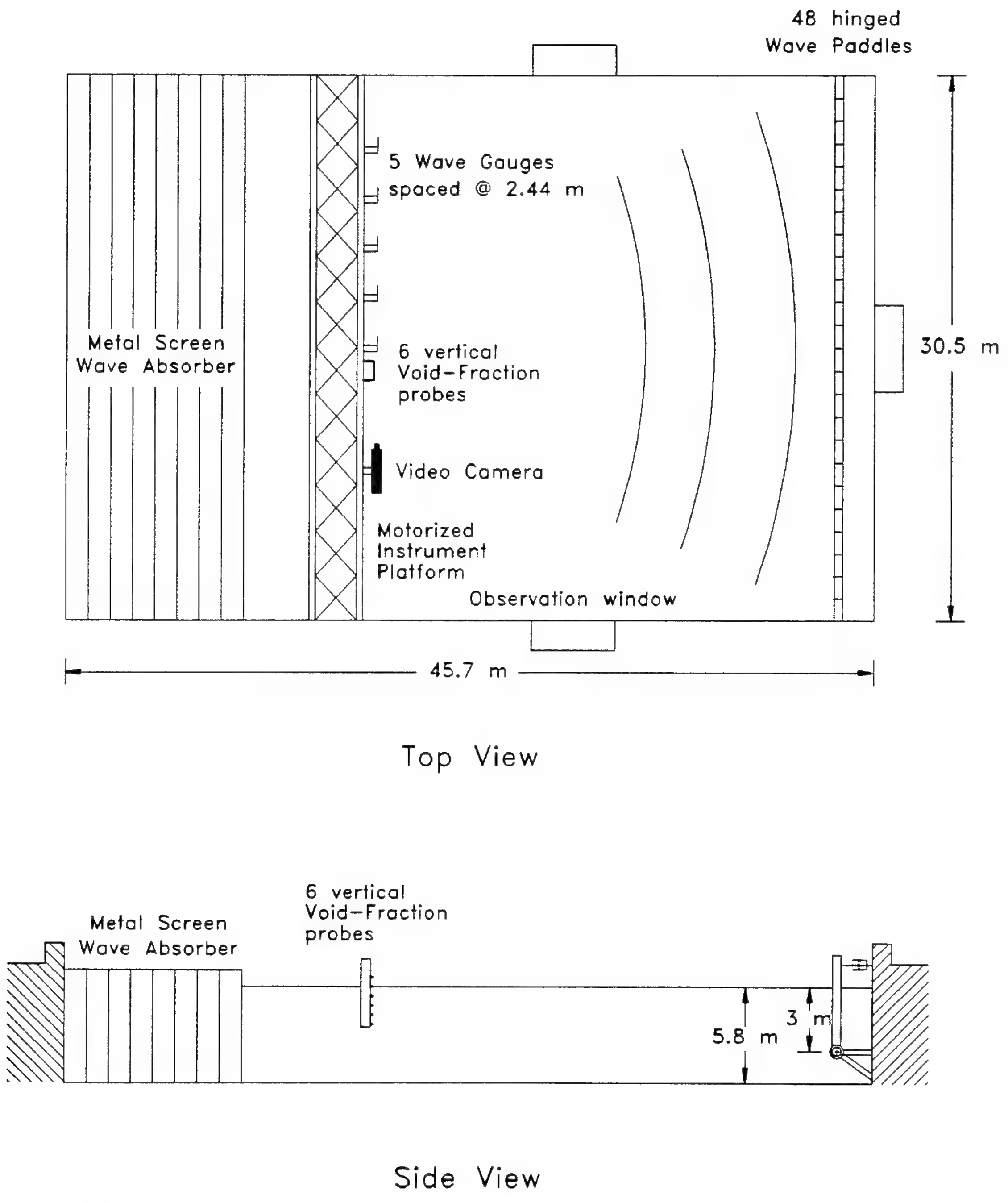


Figure 4.1: Top and side view of the multi-directional wave basin facility at the Offshore Technology Research Center (OTRC). The radial lines sketched in the top view represent the wave crests.

4.1.2 The instrumentation

The wave height was measured with five capacitance wave gauges calibrated daily. The volume fraction of air (or void-fraction hereafter) in the bubble plume generated by breaking waves was measured with a vertical array of six void-fraction probes (figure 4.2). The void-fraction in an air-water mixture can be measured by making use of the large difference in electrical conductivity between air and water. The probes used were identical to the ones described in §1.

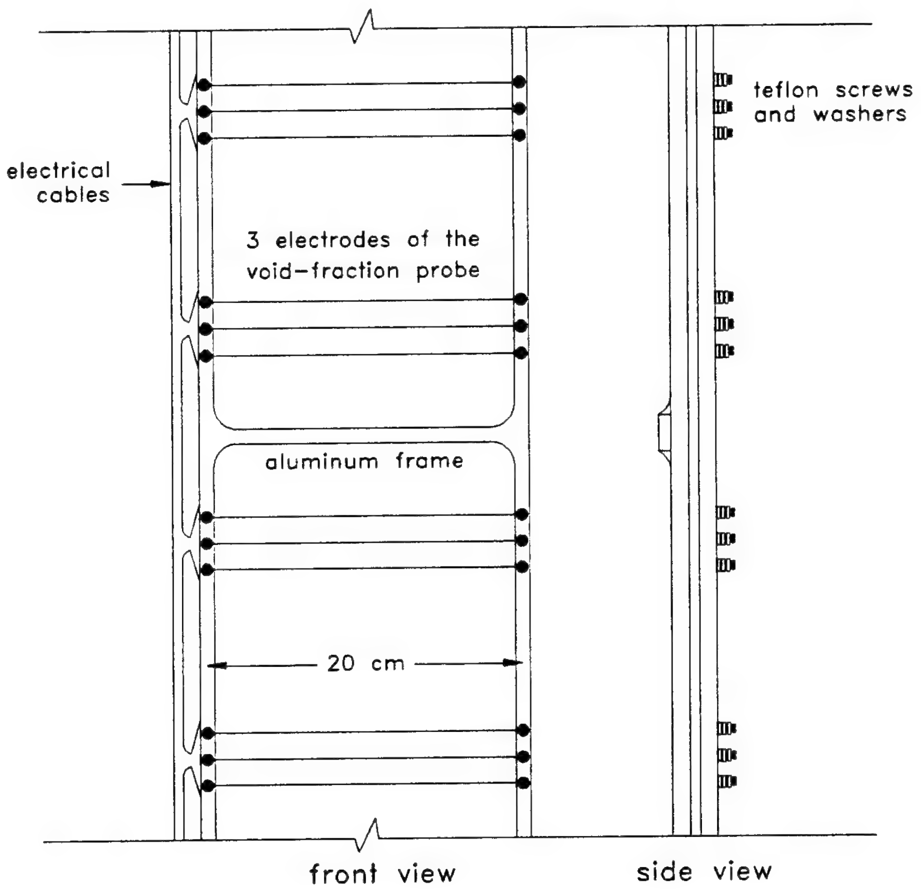

teflon screws and washers
electrical cables
3 electrodes of the void–fraction probe
aluminum frame
20 cm
front view
side view

Figure 4.2: Front and side views of a section of the array of void-fraction probes used to measure the volume fraction of air. Each probe is made up of three Nichrome wire electrodes 0.127mm in diameter. The wire separation is 1.6cm. The measuring volume for each probe is roughly a cylinder ~5cm in diameter and 20cm in length centered around the midle electrode. The separation between the probes is 15cm. Details of the characteristics of the probe and its calibration are given in §2.

Video cameras equipped with a time-code generator were used to visualize the bubble plume both from above and below the water surface. Two hydrophones and a microphone were deployed close to the breaking waves in order to measure the acoustic signature. Refer to Loewen and Melville [1993] for the details of the acoustic results.

4.2 Experimental procedure

The experiment consisted of measuring the void-fraction field in a cross-section of the bubble plume located at the centerline of the basin for three different amplitudes of the same wave packet. The amplitude of the packet was controlled through a gain factor G applied to the signal fed to the wavemaker. Table 4.1 gives the important wave parameters for the three waves studied.

G	t_b (s)	t_d (s)	x_b (m)	H (m)	V_o (cm^2)	r_o (cm)	E_o (J/m)	Δx (cm)	Δz (cm)	Δx_m (cm)	Δz_m (cm)
0.70	10.45	10.60	16.7	0.783	217.4	8.3	29.4	10.2	7.5	2.0	1.5
0.55	10.20	10.60	15.9	0.704	138.1	6.6	18.0	10.2	7.5	1.1	1.1
0.40	10.35	10.75	15.2	0.534	73.7	4.8	7.0	7.6	5.0	1.1	1.0

Table 4.1: Characteristics of the three wave packet amplitudes studied. G is the gain of the signal and it is used here to identify the wave packet. The time from initial motion of the paddles up to breaking is t_b which is taken as the time when the forward-moving jet of the breaker strikes the free surface at a distance x_b from the paddles. Note that t_b is also the time when the underwater acoustic signature from the breaking wave begins. The earliest time when complete maps of the void-fraction field are available is t_d. The maximum wave height at $x = x_b$ is H. V_o and E_o are the volume of air entrained by and the potential energy of the bubble plume per unit width at $t = t_b+0.4$s which is the earliest time when complete void-fraction maps are available for all three wave packet amplitudes. The radius $r_o = \sqrt{V_o/\pi}$. The measurements of the void-fraction were taken on a grid of size Δx by Δz. The data was interpolated on a finer grid of size Δx_m by Δz_m.

A breaking wave was generated every two minutes, and at each new position of the instrumentation three repeats of the experiment were averaged in order to reduce the inherent variance of the void-fraction signal. In our earlier 2D experiments, the void-fraction measurements were found to have good repeatability when ensemble averaged over three repeats of the breaking wave (refer to §3). For each breaking event, void-fraction and wave gauge data were acquired along with video from above and below the surface. All data acquisition systems and the video recorder were triggered from the wavemaker system in order to obtain synchronized data from one breaking event to the next.

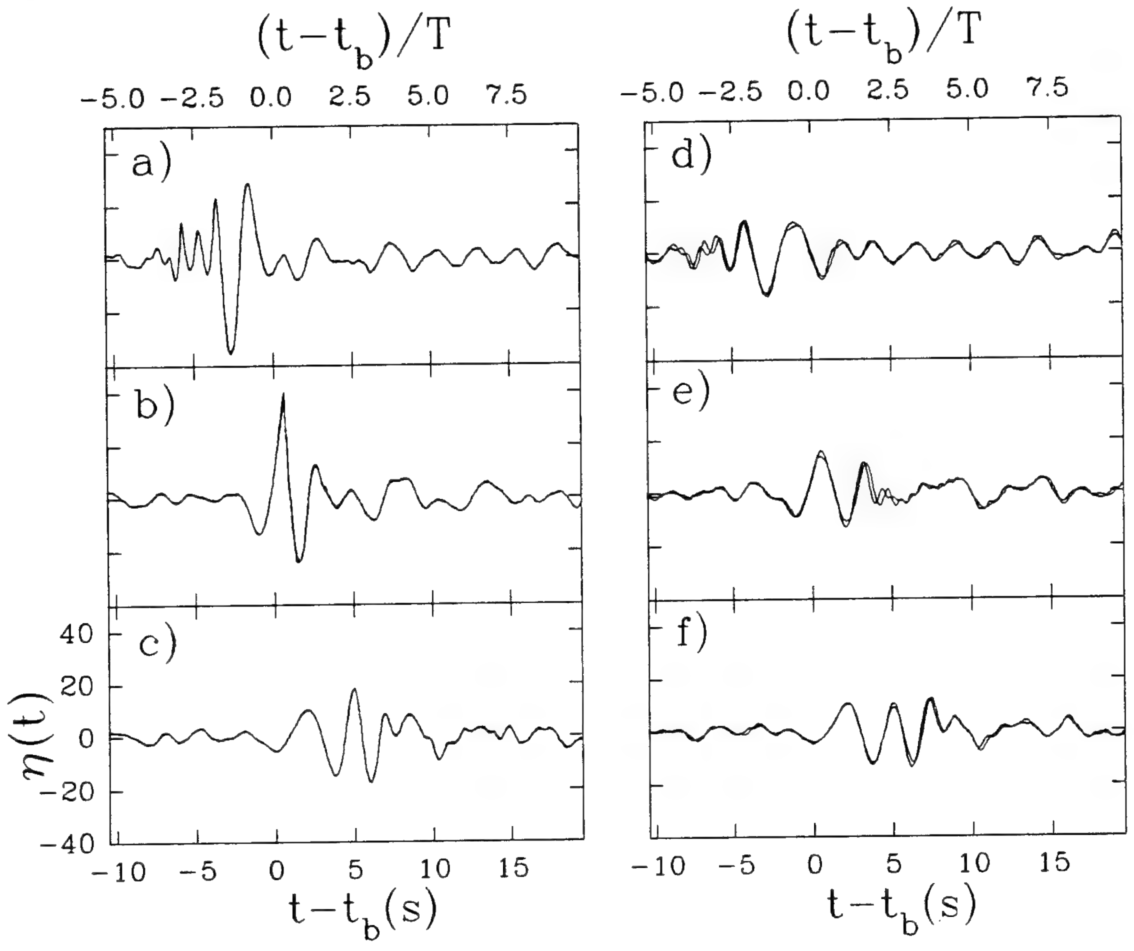


Figure 4.3: Wave gauge measurement for three repeats of the experiment at a) 6.6m upstream of breaking, b) 1.0m downstream and c) 8.6m downstream. At each position, the three plotted repeats are essentially indiscernible and the wave profile shows excellent repeatability. d,e,f) Time-series of wave gauge measurements for two wave gauges placed symmetrically at 4.9m on either side of the basin's centerline at d) 6.6m upstream of breaking, e) 1.0m downstream and f) 8.6m downstream. The figure clearly shows that the wave profile was symmetric with respect to the centerline. The time t-t_b corresponds to the beginning of breaking and T is the characteristic period of the wave packet. All time-series are for a gain G=0.7.

4.3 Description and measurement of the wave field

4.3.1 Wave height measurements

Approximately 300 repeats of the same breaking wave were necessary in order to map the void-fraction field in the central cross-section of the bubble plume. A two minute repetition cycle was found to be sufficient to allow for the water surface to come to rest between each breaking event. Figure 4.3 shows wave gauge time series upstream and downstream of breaking for three repeats of the experiment (i.e. three curves are shown on each plot). The time $t=t_b$ corresponds to the beginning of breaking when the forward moving jet strikes the free surface. The characteristic period of the wave packet is T=2sec. These time series show excellent repeatability of the wave profile both upstream and downstream for the full 30sec shown. The symmetry of the wave packet with respect to the centerline of the basin was also checked by positioning two wave gauges at 4.9m on either side of the centerline. Figure 4.3 shows time series for these two gauges both upstream and downstream of breaking. Slight phase and amplitude differences can be observed but overall the wave packet was symmetric.

Time series measured at six different locations separated by 3m along the centerline are shown in figure 4.4a-f. The chirp-like nature of the wave packet can be observed in figure 4.4a where the high frequency waves precede the longer low frequency waves. The wave packet is well focused in figure 4.4c at the onset of breaking. Wave height spectra for the time-series in figure 4.4a and figure 4.4f are shown in figure 4.5. The center frequency of the wave packet and the bandwidth are approximately 0.5Hz and 1Hz respectively. As expected, the downstream spectrum (dashed line) shows a decrease due to dissipation caused by breaking and radial spreading after breaking.

Estimates of the energy dissipation due to breaking have been difficult to make. The energy dissipation per unit width (along the wave crest) in the breaking region is expected to be comparable to the 2D case [Rapp and Melville, 1990] but, when averaged over the full width of the tank, it is much smaller since the ratio of breaker width (along the wave crest) to tank width is approximately 0.1 for the most energetic breaking wave studied. Therefore, wave gauge measurements must be taken on a very fine grid in order to resolve accurately the wave energy flux as it propagates downstream, and the wave absorbing beach must prevent all reflections.

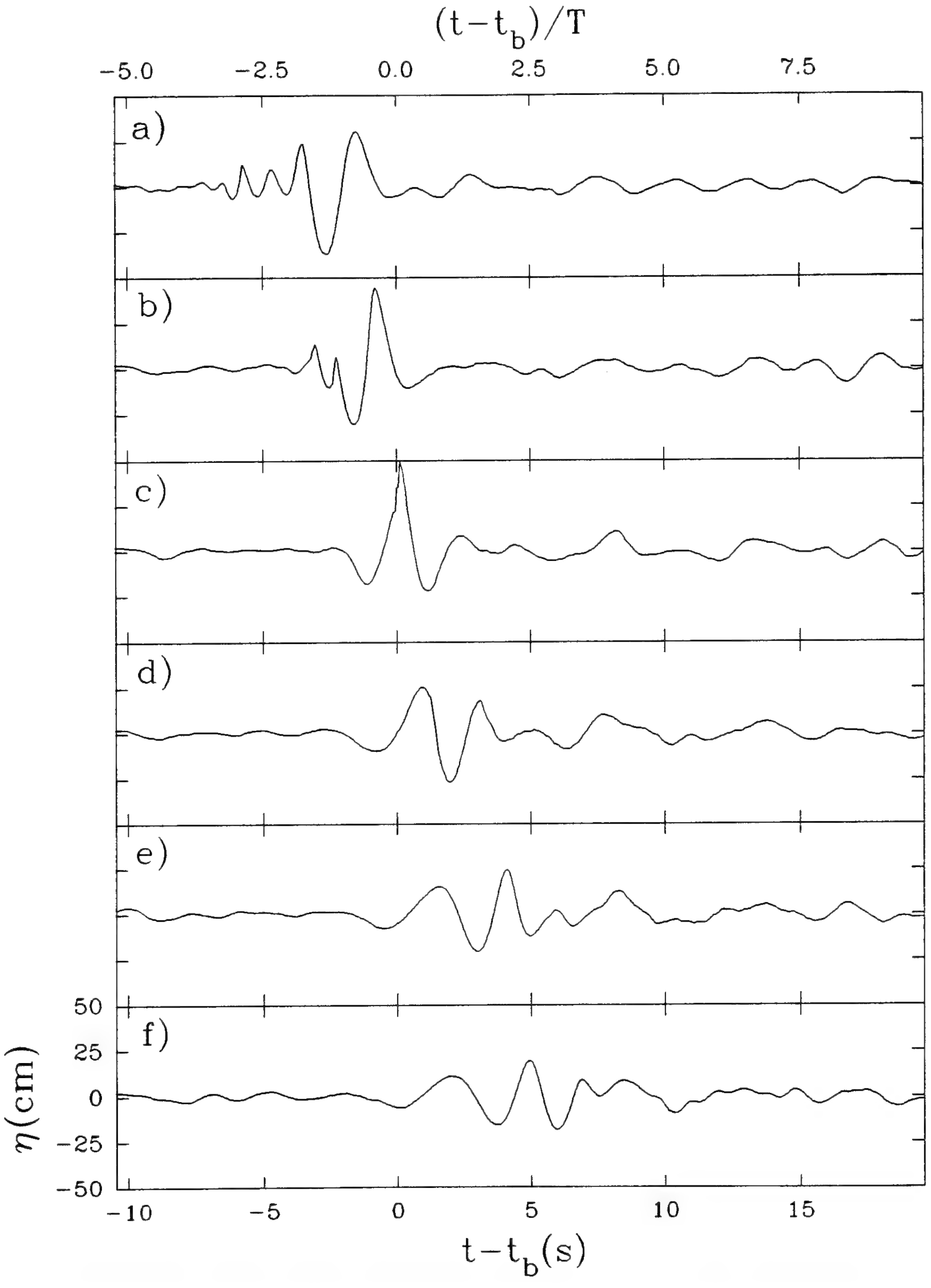


Figure 4.4: Wave gauge time-series measured at six different locations separated by 3m along the basin centerline. Breaking occurs at frame c). All time-series are for G=0.7.

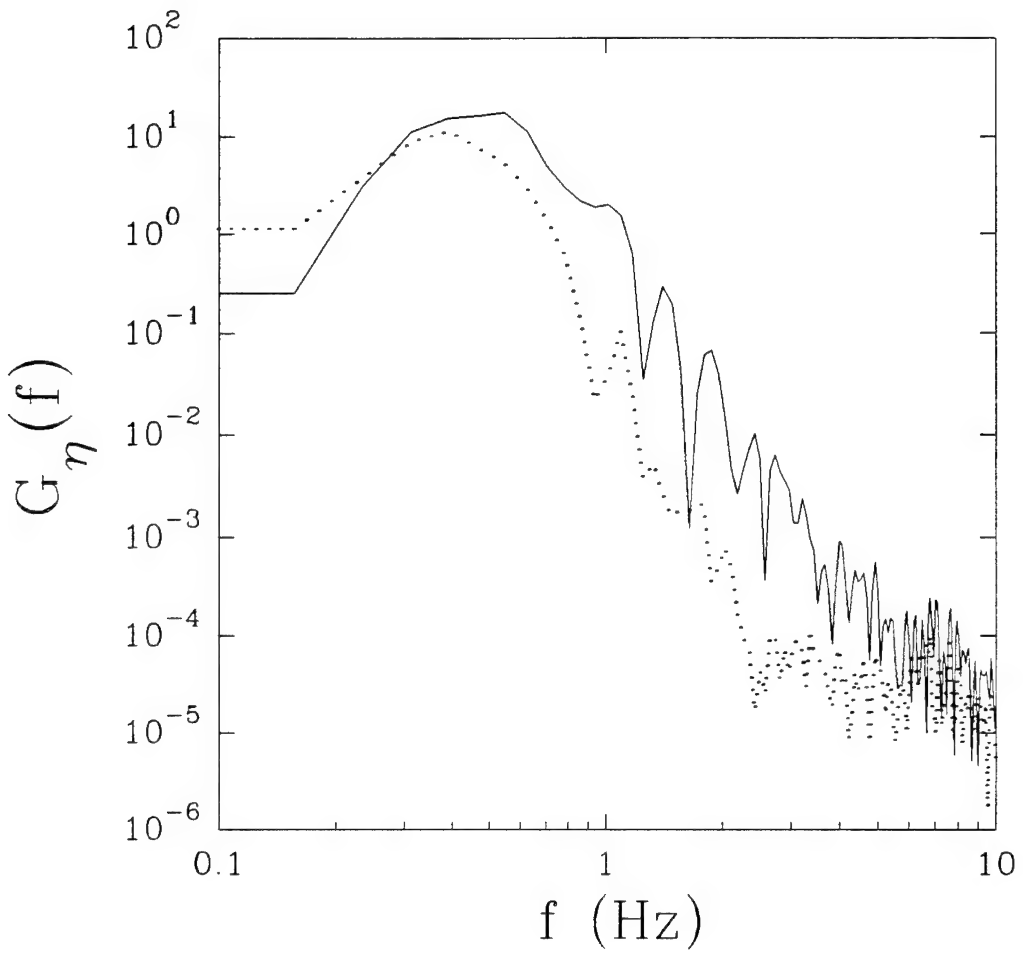
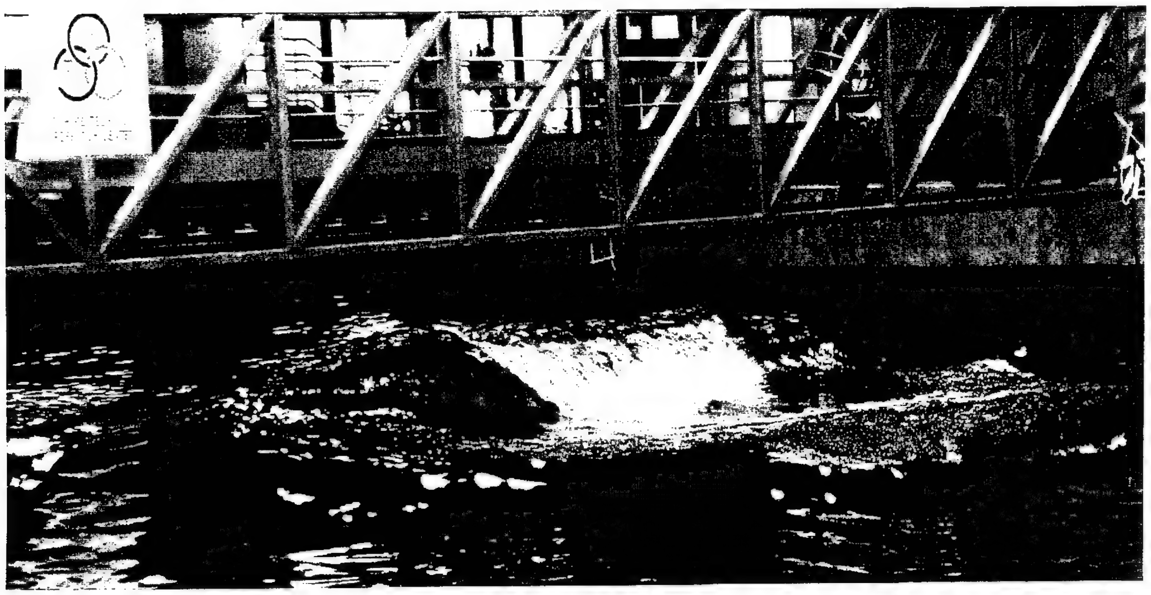


Figure 4.5: Spectra of the time-series shown in figure 4.4a (solid line) and figure 4.4f (dashed line).

4.3.2 Qualitative differences between 2D and 3D breaking waves

As the 3D wave progresses toward breaking, the central section of the wave crest plunges forward first and breaking continues by spreading laterally. This effect is depicted in figure 4.6a which shows a photograph of a breaking wave taken at the OTRC facility. Figure 4.5b shows a similar photograph taken during the Surface Wave Dynamics Experiment (SWADE) off the coast of Delaware in a water depth of approximately 2400m. Figures 4.5a and figure 4.5b have similarities with a characteristic forward plunging jet starting at the center of the wave crest where it is steeper and progressively breaking from the center to the edges.



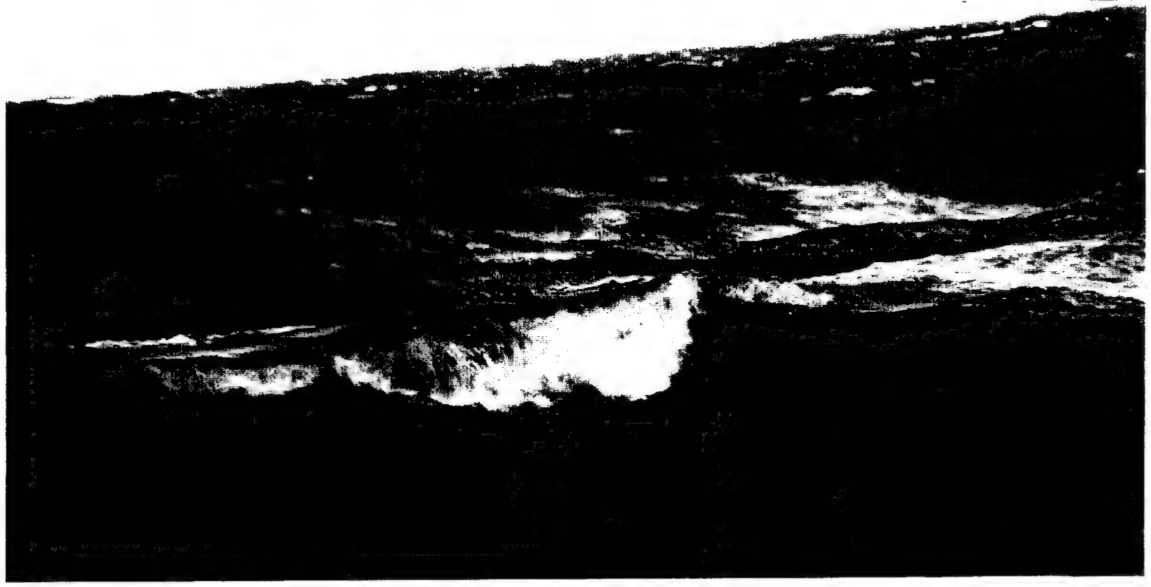


Figure 4.5: a) Breaking wave at OTRC (Photograph by R.P. Johnson of OTRC). b) Breaking wave photographed in deep water during the Surface Wave Dynamics Experiment off the coast of Delaware. Both waves begin to break at the center of the wave crest and progressively break from the center towards the edges.

As is typical, for both 2D and 3D plunging waves, a large bubble plume is produced along with a secondary plume generated by the splash from the initial impact at breaking.

The secondary plume is generally shallower and is located downstream of the primary plume. In the 3D case, we have observed less penetration of the secondary plume even thought the elevation of the splash in both the 2D and 3D cases were comparable (and of the order of the wave amplitude at breaking). One possible explanation for this effect is that in the 3D case the splash is not only projected forward but also laterally. This is evident in figure 4.7 which shows the whitecap coverage for one of the breaking wave.

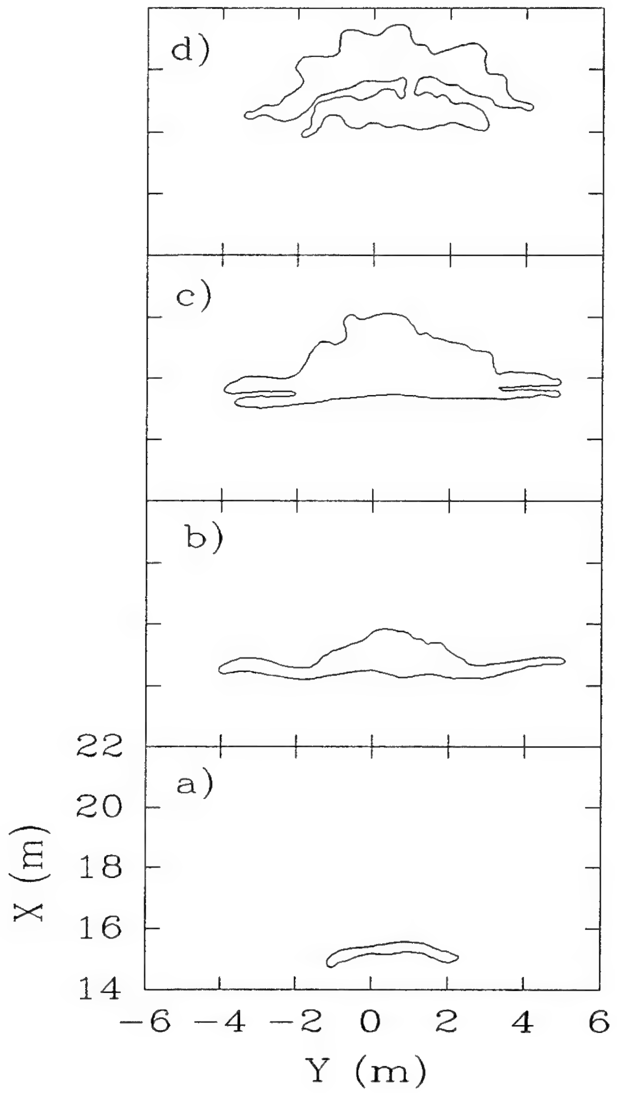


Figure 4.7: Evolution of the whitecap coverage for one of the breaking waves at OTRC (G=0.7). The frames are separated by 0.27s (or 0.135T). The figure was digitized from video images obtained with a camera looking from above. In d), the width of the secondary bubble plume (upper area in d) is greater than the width of the primary bubble plume (lower area in d).

The splash water generated at impact lands over a greater area in the 3D case because of this lateral spreading effect. Since there is less water falling onto the surface per unit surface area, there is accordingly less penetration of the secondary bubble plume being generated by this falling water. Finally, the last qualitative feature of 3D breaking waves we wish to show is the characteristic curvature of both the primary and secondary bubble plumes as depicted in figure 4.7. This feature is due to the fact that the central part of the breaker not only plunges forward first but it also plunges further downstream followed progressively by lateral sections which break later in time and upstream in space.

4.4 Void-fraction measurements

Typical void-fraction time-series obtained using the array of void-fraction probes are shown in figure 4.8. Figure 4.8a shows a time-series of the wave amplitude obtained from a wave gauge located at the same position as the vertical array of void-fraction probes. Figures 4.8b-g show void-fraction time-series for probes separated vertically by 15cm with the uppermost probe (figure 4.8b) being located at the still water level. The large "step like" transitions in figure 4.8b and figure 4.8c are due to the crossing of the water surface by the void-fraction probes. The data were edited to eliminate these transitions (refer to §2 and §3 for a discussion on "surface effect"). Video imaging of the void-fraction probes helped in this task. Notice how the start of the void-fraction signal at each probe in figure 4.8 is slightly delayed as the plume moves downward from one probe to the next.

The void-fraction measurements were made on a grid spacing of Δx by Δz which are given in Table 4.1. Void-fraction measurements at each position on this grid were ensemble averaged over three repeats of the experiments to reduce the inherent fluctuations of the void-fraction measurements (this technique is similar to the one described in §2). Spatial maps of the void-fraction field were reconstructed from the ensemble averages. Raw data was time averaged over a 0.05s window and interpolated on a finer grid of size Δx_m and Δz_m (see table 4.1) using a Laplacian and spline scheme (§3). The inherent assumption behind this interpolation process is that the ensemble averaged void-fraction field varies smoothly across the bubble plume. Void-fractions below 0.3% were assumed to be below the detection threshold of the instrument. Figures 9a,b,c,d show a sequence of color contour maps of the void-fraction field for the wave with a gain G=0.7. The color scale is linear and increases in steps of 6% void-fraction from 0.3% up to 54.3%. At early times, the data show sharply decreasing void-fractions from the center of the plume out to the edges. At later times, the void-fraction field is more uniform due to the fast degassing of the plume caused by bubbles rising back to the surface, and mixing of the plume with the ambient fluid.

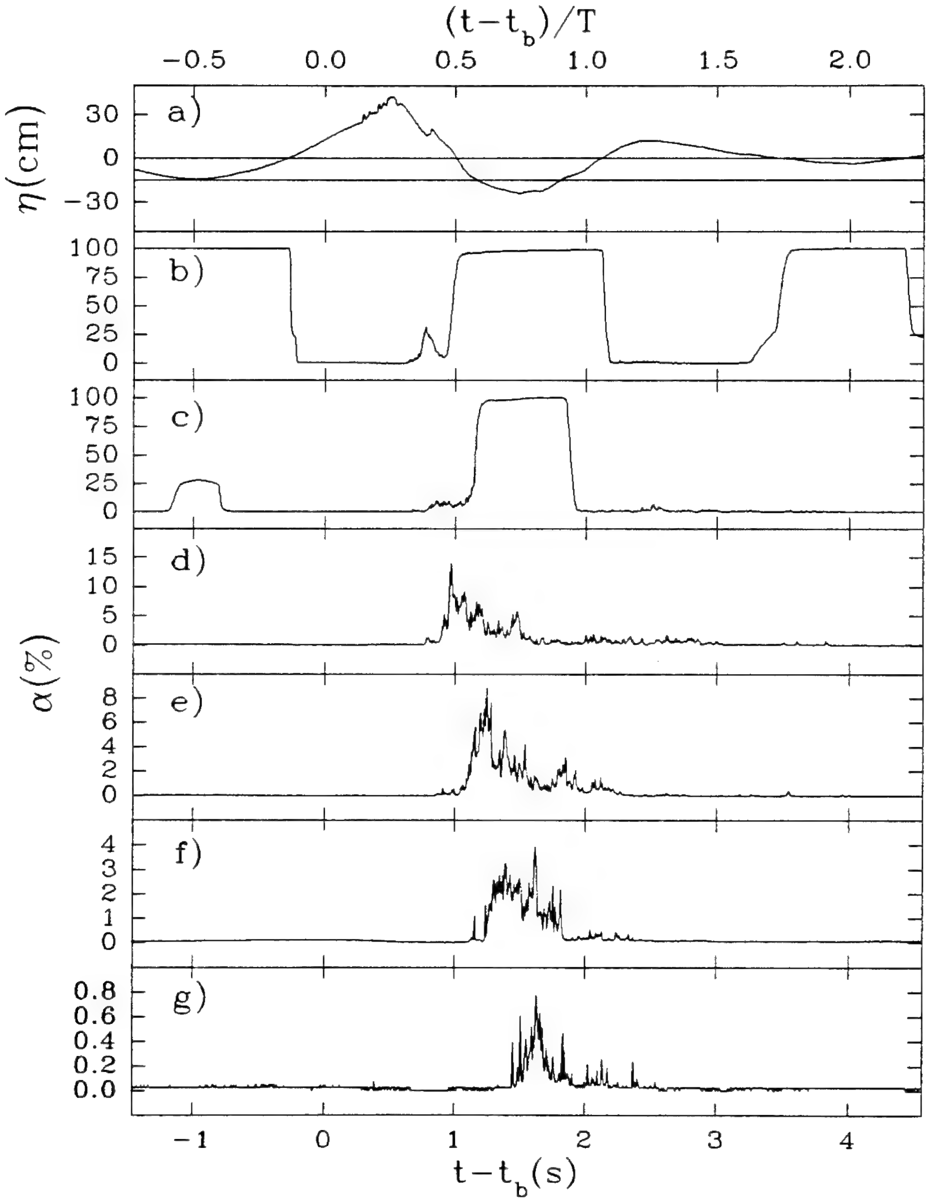


Figure 4.8: Typical time-series at 200Hz of a) wave gauge measurement; b-g) void-fraction measurements at depths 0, -15, -30, -45, -60 and -75cm respectively. Notice the different void-fraction scales for the various plots. Note how the probes cross the water surface in b) and c) and generate large "step like" features in the void-fraction time-series. The probe in c) does not quite cross the water surface during the first wave trough. All time-series are for G=0.7.

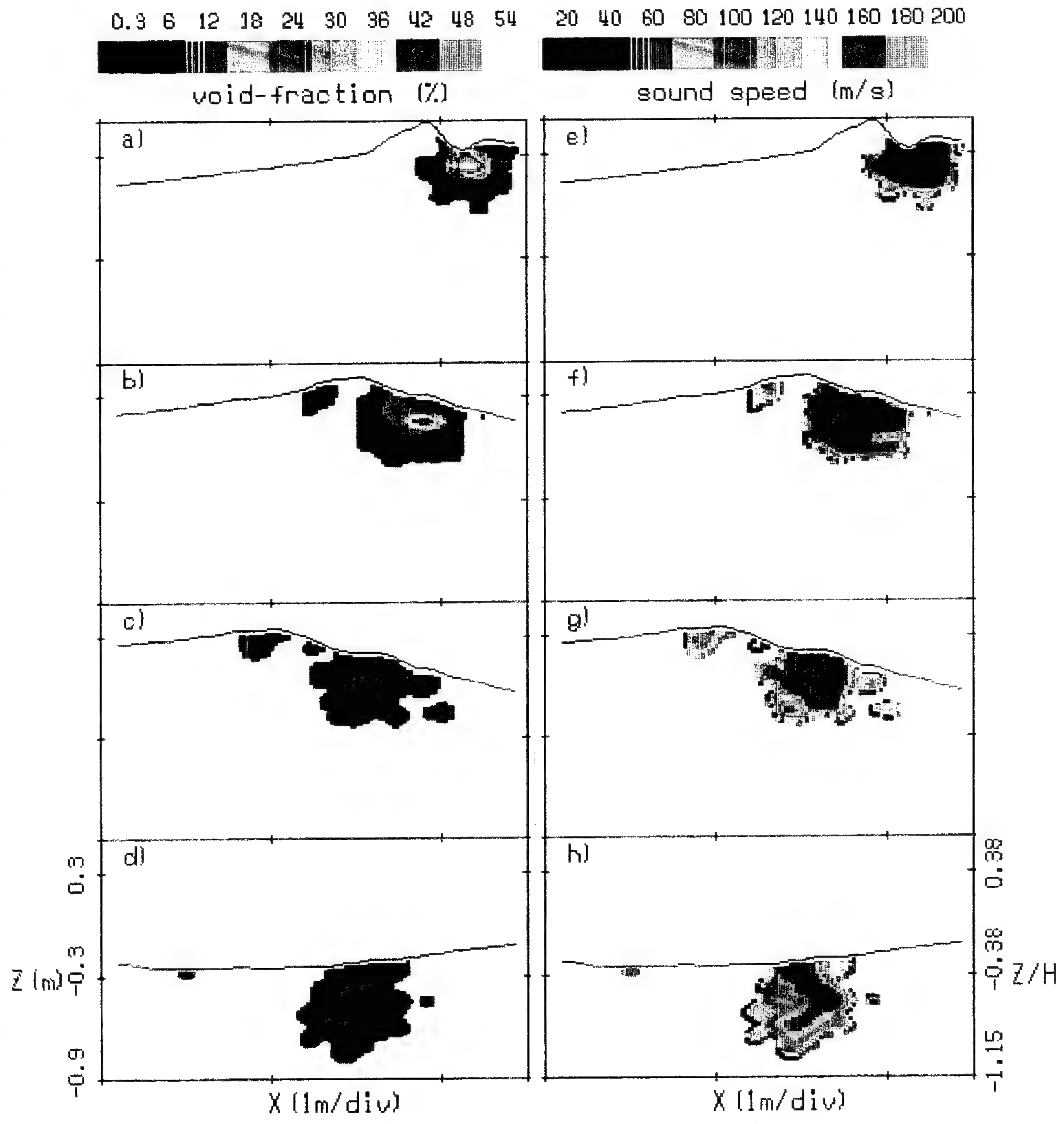


Figure 4.9: a),b),c),d) Color contour maps of the void-fraction field. The wave moves from right to left and the corresponding times t-t_b are a) 0.15T, b) 0.25T, c) 0.35T, d) 0.8T. The void-fraction color scale increases by uniform increments of 6% starting at 0.3%. e),f),g),h) Corresponding color contour maps of the sound-speed field. The sound-speed color scale increases by increments of 20m/s and corresponds to the range covered by the void-fraction scale. The breaking wave shown is for G=0.7.

Equivalent maps for the sound-speed within the bubble plume are shown in figures 4.9e,f,g,h where the void-fraction is directly converted to sound-speed using Wood's equation [Wood, 1941]. The sound-speed color scale is linear and covers the equivalent void-fraction scale of figures 4.9a,b,c,d. In contrast to the void-fraction maps, the sound-speed maps at early times show important gradients on the edges of the plume. At later times, when the void-fraction in the plume has dropped to O(1%), we find important sound-speed gradients throughout the plume. These differences are obviously due to the character of Wood's equation [Wood, 1941; appendix A].

4.5 Moments of the void-fraction field

From the void-fraction measurements, various moments of the void-fraction field can be computed,

$$A = \int_A dA \qquad (4.1)$$

$$V = \int_A \alpha \, dA \qquad (4.2)$$

$$\bar{\alpha} = \frac{V}{A} \qquad (4.3)$$

$$E = \rho g \int_A \alpha z \, dA \qquad (4.4)$$

where A is the total cross-sectional area of the bubble plume above a void-fraction threshold of 0.3%; V is the volume of air entrained per unit width; α is the void-fraction; E is the work per unit width required to entrain air against buoyancy; ρ is the water density; g is the acceleration due to gravity; z is the depth from the water surface and $\bar{\alpha}$ is the average void-fraction. It should be pointed out that V and E are volume and potential energy of the plume per unit width along the crest of the breaking wave.

In order to compare the results of the 3D experiments with those of the 2D experiments (§3), the various moments described above are normalized as follows: $V^* = V/V_o$, $A^* = A/V_o$ and $E^* = E/E_o$ where $V_o = V(t=t_b+0.4s)$, $E_o = E(t=t_b+0.4s)$ and t_b is the time at which breaking begins with the impact of the surface on itself. The specific time $t=t_b+0.4$s selected to define V_o and E_o was chosen because it was the earliest time for which complete maps of the void-fraction field were available for all three waves. Note that in the 2D case, V and A had been normalized by the volume per unit width of the 'cylinder' of air enclosed at $t=t_b$ which was measured with a video camera from the side of the channel. In the 3D experiments, such video measurements are much more difficult to realize and may not even be possible because of the curvature of the wave crest. Furthermore, it should be pointed out that the potential energy E in the 2D experiments was normalized by E_d, the total energy dissipated by breaking (refer to §3). Table 4.1 shows the values used for V_o and E_o in the 3D experiments.

Figure 4.10 shows the above moments for the three waves studied. The solid lines are the functional forms obtained in the 2D experiments (§3). The data for V^* in figure 4.10a shows that up to 0.5T, the bubble plume degasses at approximately the same rate as in the 2D case after which it assumes an appreciably slower decay rate out to 1.5T. The slower decay rate seen in the 3D experiments during the later stage of the bubble plume evolution is also a characteristic of the other moments as we will see shortly. The data for A^* (figure 4.10b) show a faster initial increase in the bubble plume cross-sectional area and a slower decrease at later times when compared to the 2D case. The plume's cross-sectional area is also consistently higher than the 2D fit (solid line). The mean void-fraction $\bar{\alpha}$ (figure 4.10c) shows decay rates comparable to the 2D case for up to 0.5T and slower rates later. The initial offset of the data with the solid line representing the 2D fit is due to the faster initial increase in cross-sectional area in the 3D case. Data for E^* (figure 4.10d) show decay rates consistent with the 2D fit up to 0.5T after which the data follows a slower rate (between 0.5T and ~1.0T) and finally a faster one again (after ~1.0T). Interpretation of the differences between the 2D and 3D results noticed at later times (after 0.5T) will be given in the discussion (§4.8).

Insight into the kinematics of the bubble plume can be gained by computing the horizontal and vertical centroids of the bubble plume

$$\bar{x} = \frac{\int_A \alpha x \, dA}{\int_A \alpha \, dA} \qquad (4.5)$$

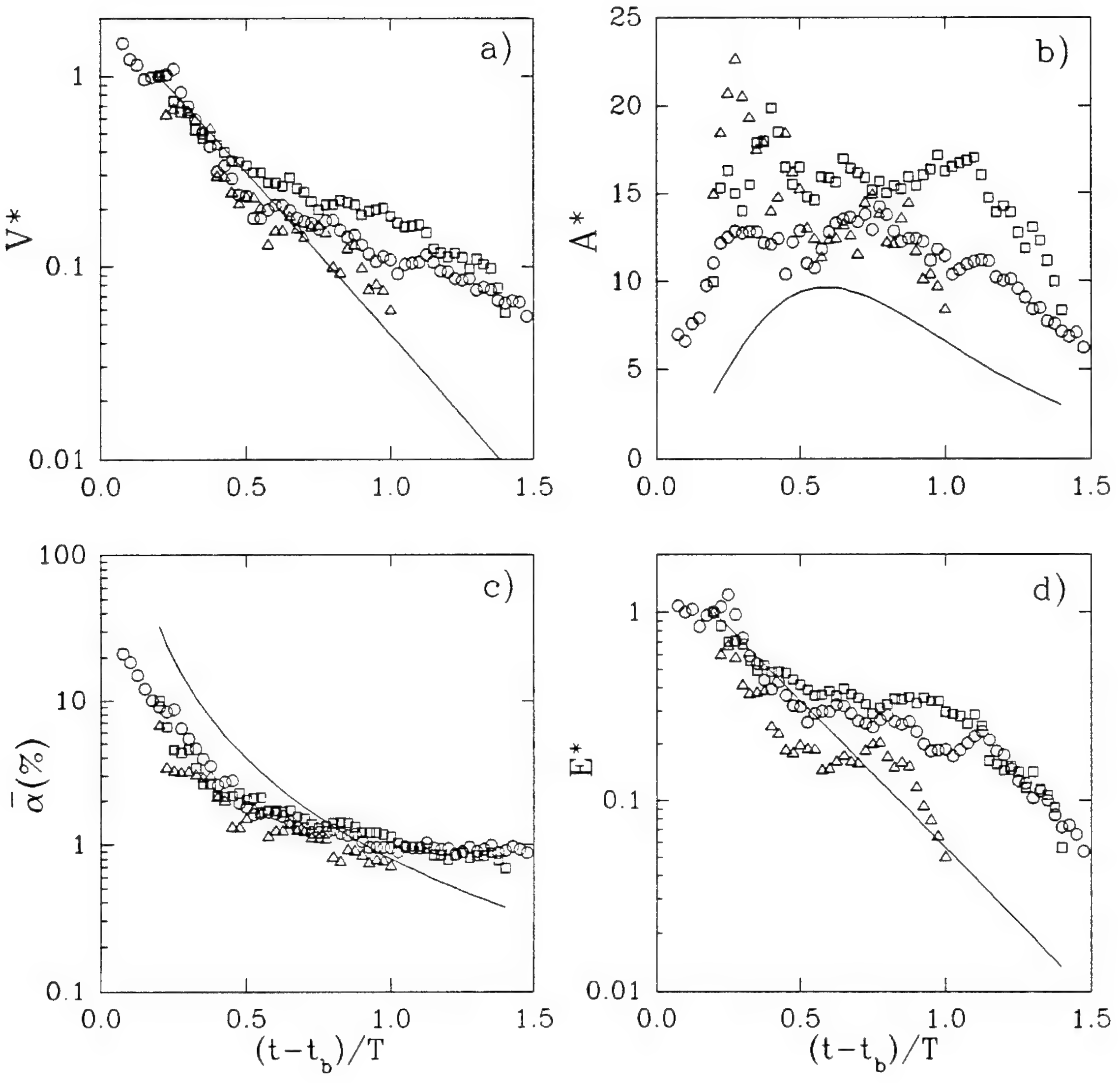


Figure 4.10: Moments of the void-fraction field as a function of time. a) Normalized volume of air entrained per unit width V*, b) normalized bubble plume cross-sectional area A*, c) mean void-fraction $\bar{\alpha}$, d) normalized potential energy E* of the bubble plume required to entrain the air against buoyancy. Solid lines are best fits from 2D experiments (refer to §3). The characteristic wave period is T=2sec. G=0.7 (O), G=0.55 (□) and G=0.4 (Δ).

$$\bar{z} = \frac{\int_A \alpha z \, dA}{\int_A \alpha \, dA} \qquad (4.6)$$

where x is the horizontal distance from x_b , the location when the forward-moving jet of the breaker strikes the free surface (i.e. at the beginning of breaking), and z is the depth from the free surface. Notice that the vertical centroid $\bar{z}$ describes the vertical position of the plume with repect to the free surface or equivalently its effective penetration depth below the water surface. Hence, the effect of the orbital velocity of the wave which will cause the plume to move up and down with each wave crest and wave trough is not included in this moment. However, the amount of vertical excursion caused by the wave can easily be estimated from the wave height measurement at breaking given in table 4.1.

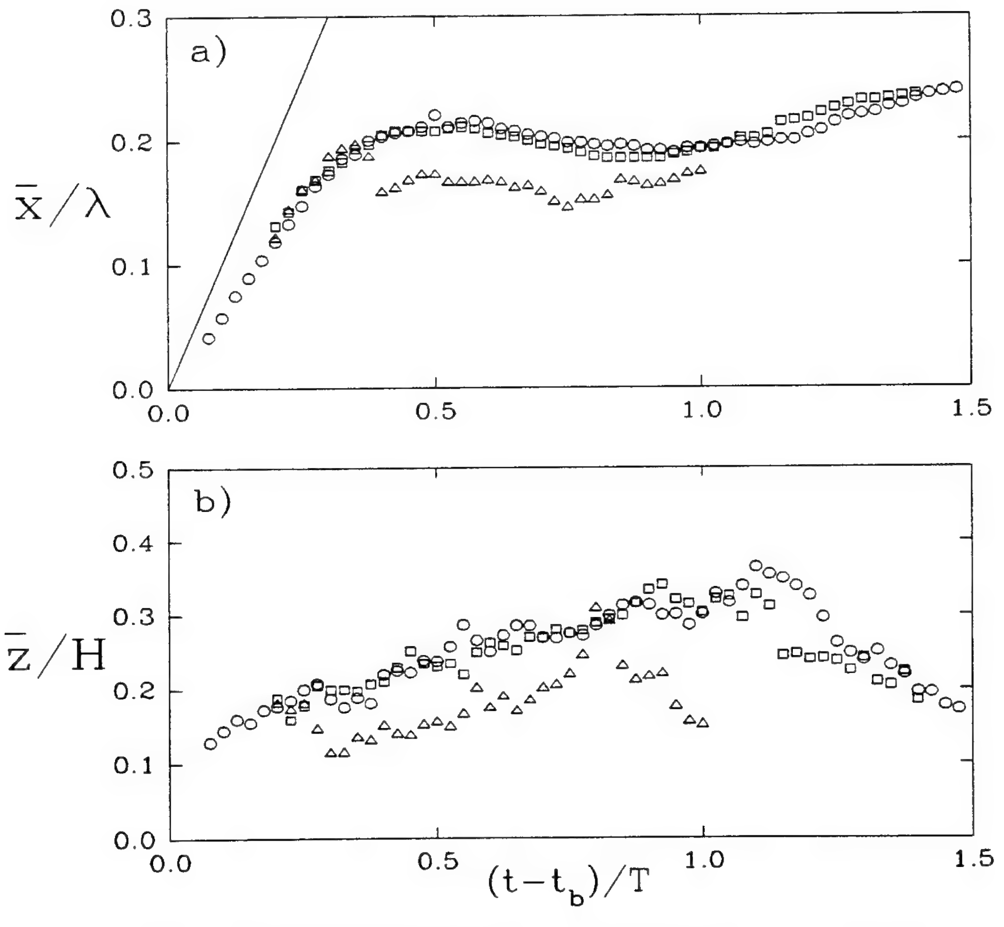


Figure 4.11: a) Horizontal centroid $\bar{x}$, and b) vertical centroid $\bar{z}$ of the bubble plume normalized by the wave length (λ=6.3m) and the wave height at breaking H (see table 4.1) respectively. Solid line is the phase speed of the wave $C = \lambda/T$. G=0.7 (O), G=0.55 (□) and G=0.4 (Δ).

Figure 4.11a shows the horizontal centroid normalized by λ=6.3m, the characteristic wavelength of the wave packet. The bubble plume moves quickly downstream to approximately 0.2λ in about 0.3T. The horizontal velocity of the plume immediately after breaking is approximately 0.7C where C=λ/T is the phase speed of the wave. The local minimum is due to the reversal in the orbital velocity at the trough of the wave which occurs approximately one wave period after onset of breaking.

Figure 11b shows the vertical centroid normalized by the height of the wave at breaking H (given in Table 4.1). The vertical centroid $\bar{z}$ /H represents the mean penetration depth of the bubble plume scaled by the wave height prior to breaking. The vertical centroid in the 3D experiments moves to its maximum value of approximately 0.3H after one wave period. The speed of penetration is approximately 0.2H/T which is much slower than the initial horizontal velocity of the plume. However, this vertical velocity is sustained for a much longer period and it remains almost constant up to a wave period after breaking is initiated.

4.6 Effect of a finite void-fraction detection threshold on the estimates of the moments

The detection threshold above which the void-fraction probe gave unambiguously a signal due to air bubbles in water was 0.3% void-fraction. It is fair to ask if the moments of the void-fraction field presented in the previous section would differ if the probe had been more sensitive and capable of detection levels below 0.3% void-fraction. Such analysis was conducted by varying the detection threshold α_{th} from 100% down to 0.3% and for each new threshold, all moments presented in figure 4.10 and figure 4.11 were recomputed. The ultimate objective of this analysis was to verify that the moments reached an asymptotic value before the threshold reached 0.3% in which case it would be clear that the moments would not change significantly with a more sensitive void-fraction probe.

Figure 4.12a shows the normalized volume of air V^* as a function of α_{th}. As expected, the total volume of air entrained is insensitive to the void-fraction threshold and it appears that a threshold of 0.3% will yield unbiased estimates of the actual air volume entrained. Note that the times $t-t_b$=0.3T and $t-t_b$=0.8T in figure 4.12 were chosen to be representative of the bubble plume at an early and later stage of its evolution. For all three waves studied in these experiments, complete void-fraction maps were available for $0.2T < t-t_b < 1.0T$. The cross-sectional area of the bubble plume A^* shown in figure 4.12b is much more dependent on α_{th} and estimates at a detection threshold of 0.3% are bound to underestimate the true value. Figure 4.12c shows that the mean void-fraction is also sensitive to the threshold. This is mainly because $\bar{\alpha} = V/A = V^*/A^*$, and A^*, as it was shown above, is sensitive to α_{th}. The potential energy of the plume E^* (figure 4.12d) shows the same kind of asymptotic behavior as V^* which indicates that estimates of this moment based on a $\alpha_{th} = 0.3\%$ are unbiased.

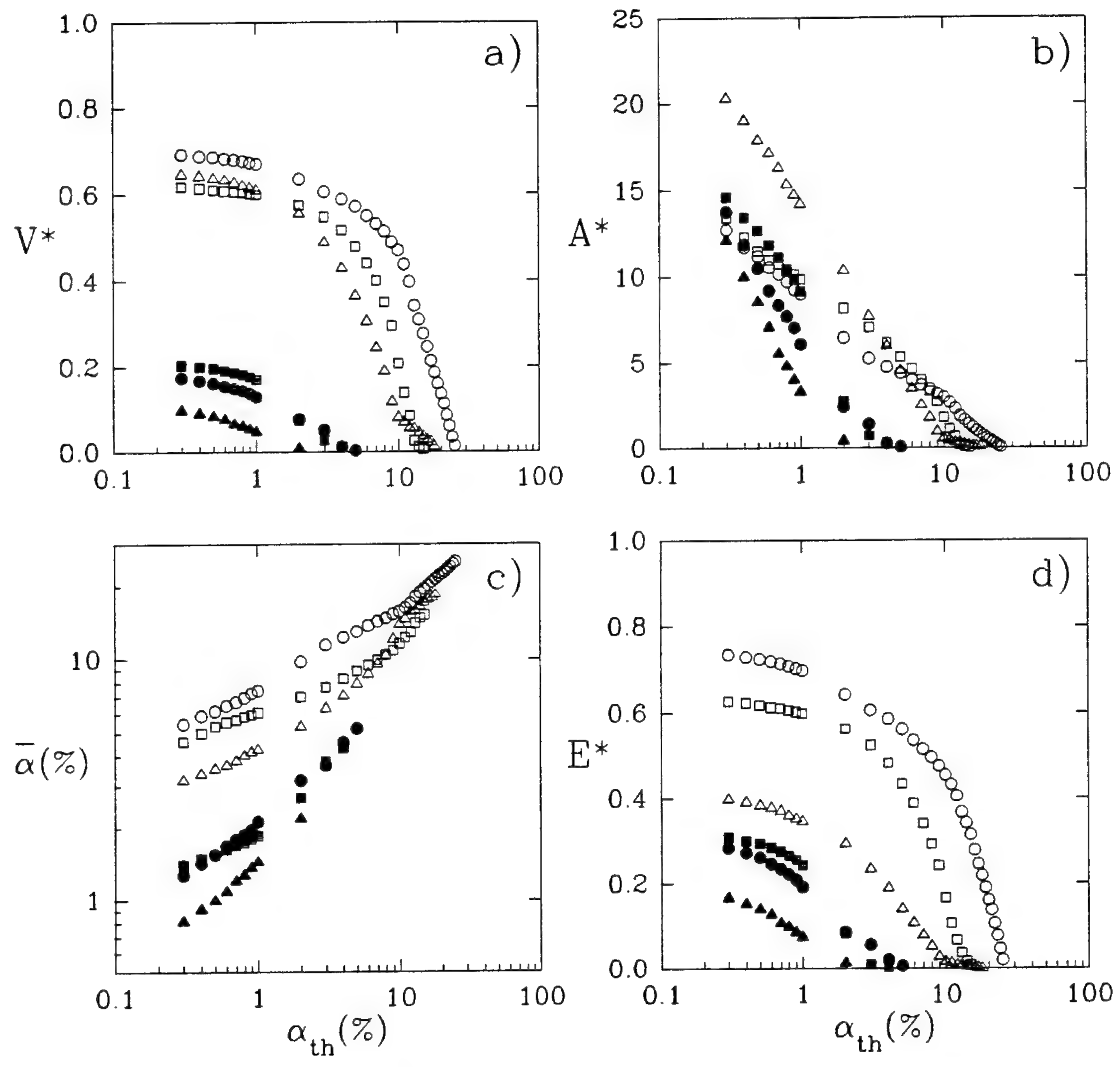


Figure 4.12: Moments of the void-fraction field as a function of the instrumentation detection threshold α_{th} for $t-t_b = 0.3T$ (hollow symbols) and $t-t_b = 0.8T$ (solid symbols). a) Normalized volume of air entrained V^*, b) normalized bubble plume cross-sectional area A^*, c) mean void-fraction $\bar{\alpha}$, d) normalized potential energy E^* of the bubble plume required to entrain the air against buoyancy. G=0.7 (O), G=0.55 (□) and G=0.4 (Δ).

The same analysis can be carried out for the horizontal and vertical centroids of the bubble plume. Figure 13 shows clearly that $\bar{x}$ and $\bar{z}$ are unbiased estimates of the true centroids since they do not show any dependence on α_{th} at small values of α_{th}.

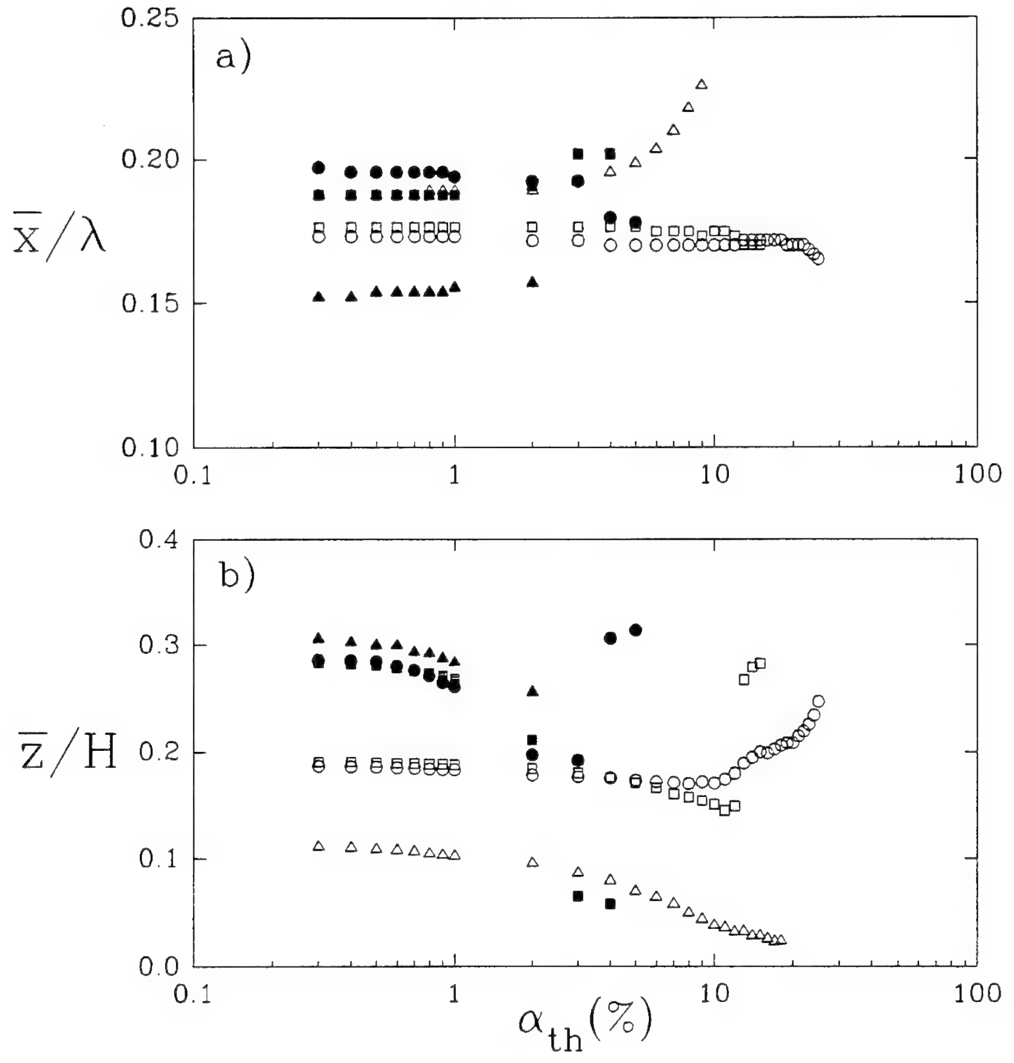


Figure 4.13: a) Horizontal centroid $\bar{x}$ and b) vertical centroid $\bar{z}$ of the void-fraction field as a function of the instrumentation detection threshold α_{th} for $t-t_b = 0.3T$ (hollow symbols) and $t-t_b = 0.8T$ (solid symbols). G=0.7 (O), G=0.55 (□) and G=0.4 (Δ).

Since the frequency f_r of the collective oscillations of a bubble plume can be estimated from the cross-sectional dimensions of the plume A and the mean void-fraction $\bar{\alpha}$, it would be important to find out if the predicted frequency of oscillation is sensitive to the threshold of the void-fraction probe. The details of the calculation of f_r are known and they are summarized in Loewen and Melville [1993] for a semicylindrical bubble plume located at the free surface. For rough estimates, the equation

$$f_r \simeq \frac{c(\bar{\alpha})}{2.5R_c} \tag{4.7}$$

will give resonance frequencies within 10% for $0.1\% \leqslant \bar{\alpha} \leqslant 10\%$ and $0.1\text{m} \leqslant R_c \leqslant 1\text{m}$ where R_c is the radius of the plume and $c(\bar{\alpha})$ is the sound-speed obtained using Wood's equation [Wood, 1941]. Figure 4.14 compares the above equation with the resonance frequencies predicted by the model of Lu et al. [1990].

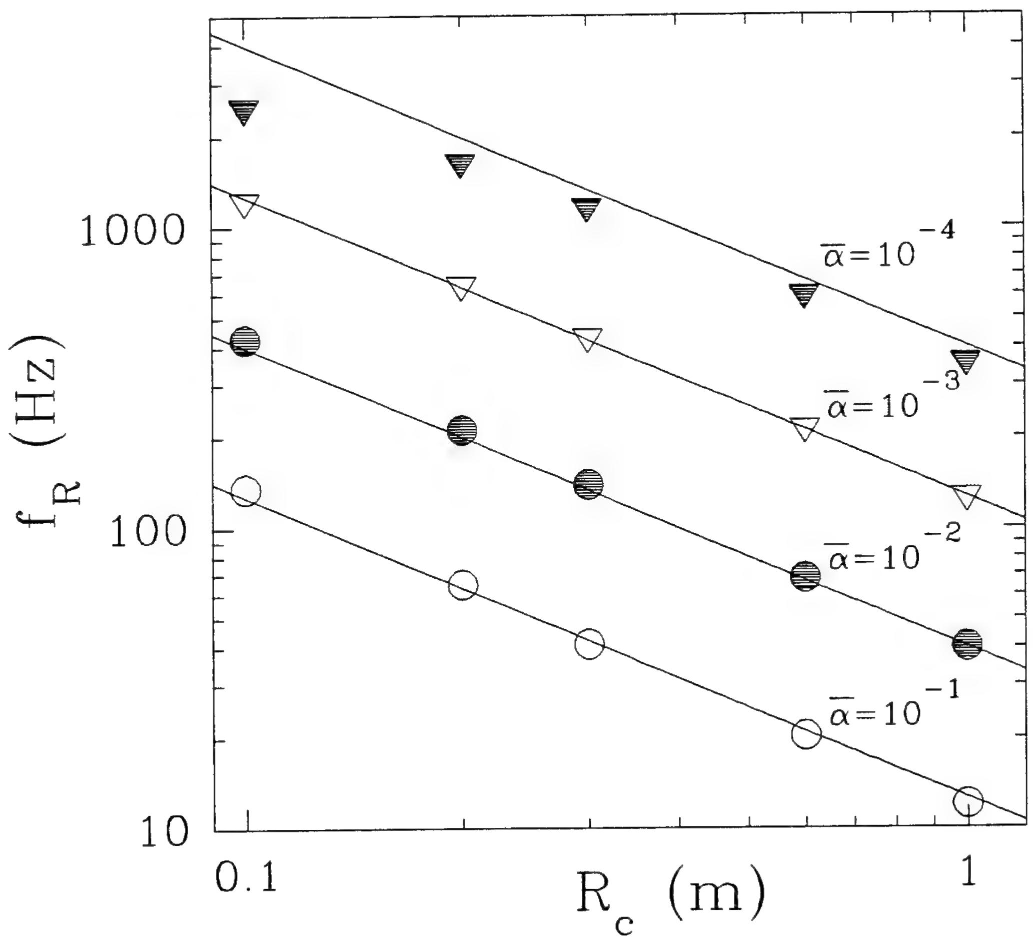


Figure 4.14: Resonance frequencies (first mode) of a semicylindrical bubble plume of radius R_c located at the free surface. The first mode is the one that behaves as a dipole and therefore it has a pressure release condition at the surface. Symbols are from a model by Lu et al. [1990]. Solid lines are from equation 4.7. From Loewen and Melville [1993].

Figure 4.15 shows that indeed, the frequency of oscillation f_r reaches its asymptotic value when $\alpha_{th} \leqslant 1.0\%$ and that below 1.0%, the computation of f_r is independent of the probe's detection threshold. This result may be surprising at first considering that $\bar{\alpha}$ and A do not reach their asymptote at a detection threshold of 0.3%. However, as the void-fraction threshold is lowered, the cross-sectional area increases. Likewise, the mean void-fraction is reduced and incidentally, the mean sound-speed increased (for $\bar{\alpha}<50\%$). The simultaneous effects of increasing A and $c(\bar{\alpha})$ on f_r balance each other out.

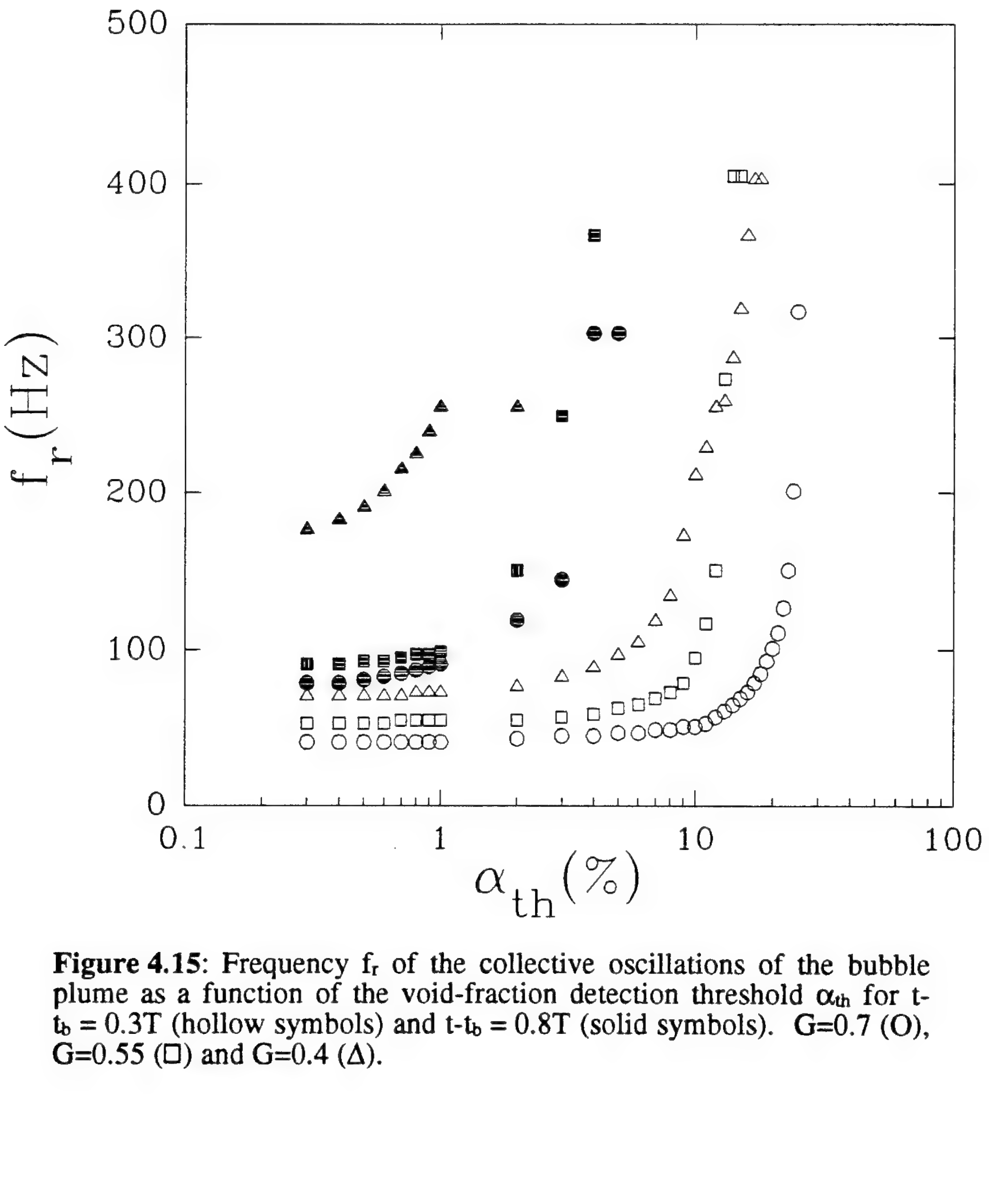


Figure 4.15: Frequency f_r of the collective oscillations of the bubble plume as a function of the void-fraction detection threshold α_{th} for $t-t_b = 0.3T$ (hollow symbols) and $t-t_b = 0.8T$ (solid symbols). G=0.7 (O), G=0.55 (□) and G=0.4 (Δ).

4.7 Radial dependence of the void-fraction and sound-speed field

From the maps of figure 4.9 the bubble plume appears relatively symmetric. Assuming that all of the void-fraction is repositioned to form an ideal equivalent semicylindrical plume without any azimuthal dependence of the void-fraction, it then becomes possible to determine an equivalent radial dependence of the void-fraction. Figure 4.16a shows the void-fraction field at $t-t_b=0.35T$ for the wave with gain G=0.7, and figure 4.16b shows the equivalent semicylindrical plume.

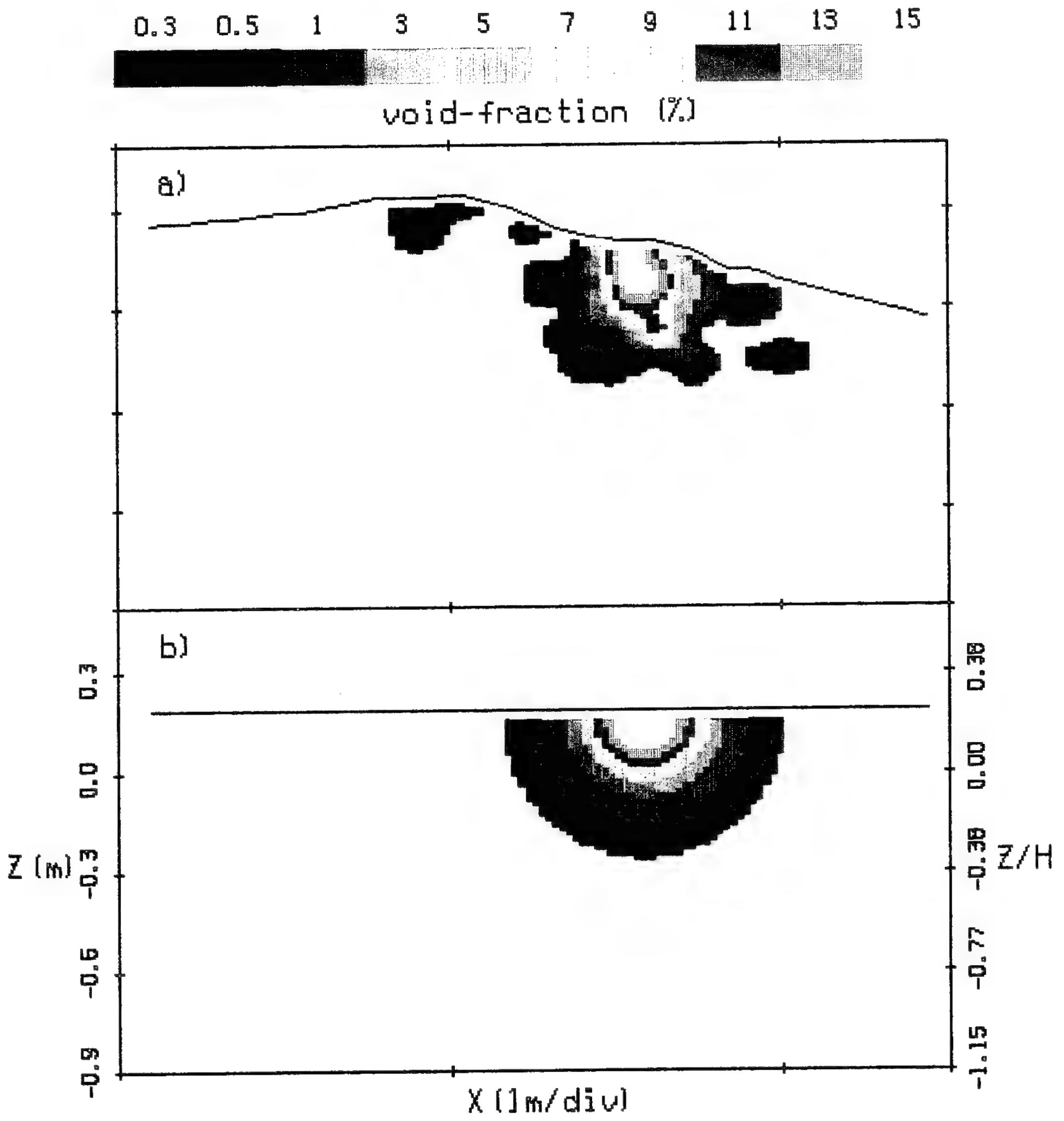


Figure 4.16: a) Color contour maps of the void-fraction field for $t-t_b=0.35T$ (G=0.7). b) Equivalent semicylindrical plume of the color map shown in a).

The equivalent semicylindrical color map of figure 4.16b is assembled from the original color map (figure 4.16a) by repositioning the elements of void-fraction such that the area over any two limits α_a and α_b

$$\int_{\alpha_a}^{\alpha_b} dA$$

is conserved between the original distribution and the equivalent distribution.

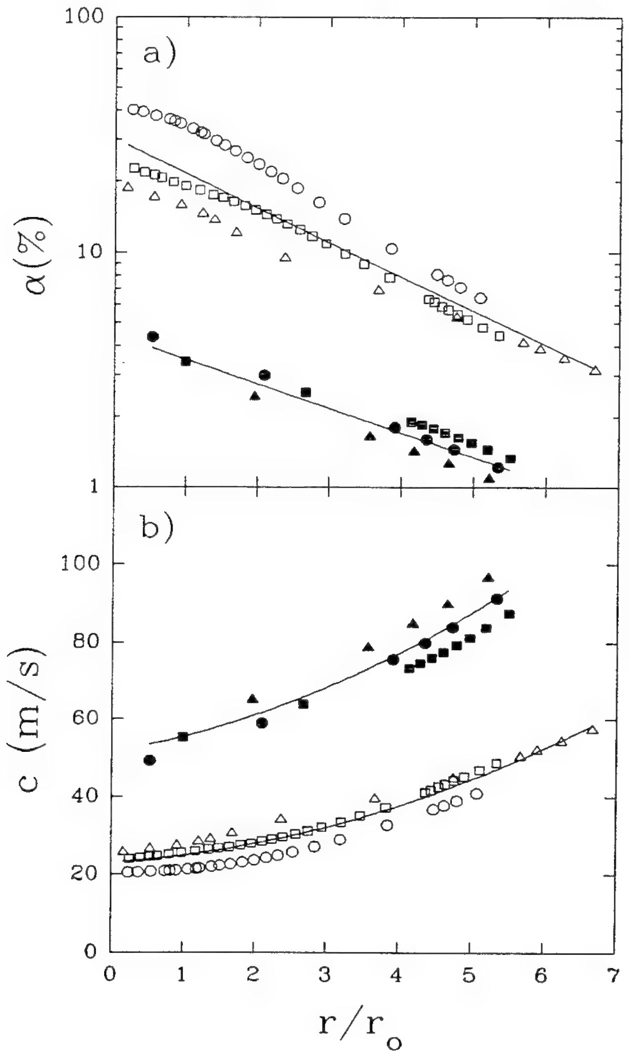


Figure 4.17: a) Void-fraction and b) sound-speed radial profiles for $t-t_b = 0.3T$ (hollow symbols) and $t-t_b = 0.8T$ (solid symbols). G=0.7 (O), G=0.55 (□) and G=0.4 (Δ). Solid lines are best fits based on a) equation 4.8 and b) equation 4.9. The radius is normalized by $r_o=\sqrt{v_o/\pi}$.

Figure 4.17a,b shows how the void-fraction and sound-speed vary as a function of the radius for two specific times, $t-t_b=0.3T$ and $0.8T$. The radius of the semicylindrical plume r has been normalized by $r_o = \sqrt{V_o/\pi}$ where V_o has been used to normalize V and it is given in table 4.1. A simple exponential fit of the form

$$\alpha(r,t) = A_0(t) \exp\left(A_1(t)\left(\frac{r}{r_o}\right)\right) \tag{4.8}$$

has been performed on the void-fraction data for all three wave amplitudes for times up to 0.8T.

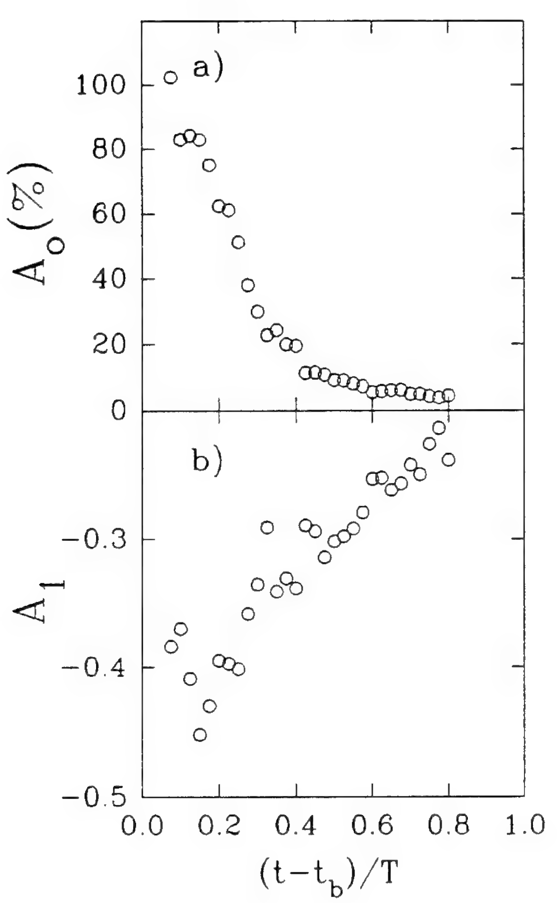
The coefficients A_0 and A_1 are given in figures 4.18a,b as a function of time. As expected, the constant term A_0 (with respect to r), which represents the void-fraction in the center of the plume, decreases rapidly shortly after breaking when the degassing of the plume is very rapid. The steepness of the void-fraction gradient described by equation 4.8 is governed by the coefficient A_1. As shown in figure 4.18b the sharpest gradients are observed immediately after the onset of breaking. As breaking evolves, the plume degasses, entrains ambient fluid and expands. The final result of this process will be to significantly decrease the void-fraction gradient as time evolves. Void-fraction radial profiles fitted with equation 4.8 have standard deviations below $0.15A_0$.

The same analysis can be carried out for the sound-speed radial profiles of figure 4.17b. The second-order polynomial fit is

$$c(r,t) = B_0(t) + B_1(t)\left(\frac{r}{r_o}\right) + B_2(t)\left(\frac{r}{r_o}\right)^2 \tag{4.9}$$

where B_0, B_1 and B_2 are functions of time and are given in figures 4.18c,d,e for times ranging from 0.15T to 0.8T. The sound-speed in the center of the plume is given by $B_0(t)$ which shows an initial decrease up to 0.25T corresponding to the void-fraction dropping from 100% down to 50% (see $A_0(t)$ in figure 4.18a) followed by a steady increase after 0.25T. The linear coefficient B_1 represents the slope of the sound-speed profile $\partial c/\partial r$ at $r=0$. Figure 4.18d shows that B_1 is initially negative which means that the sound-speed decreases as r increases in the neighborhood of $r=0$. After $t-t_b=0.25T$, B_1 becomes positive and increases with time indicating that the slope of the sound-speed profile near $r=0$ is positive and becomes steeper as time evolves. The positive values for B_2 (figure 4.18e) at all times imply that the sound-speed profile is concave upwards.

Sound-speed radial profiles fitted with equation 4.9 have standard deviation below $0.15B_0$.



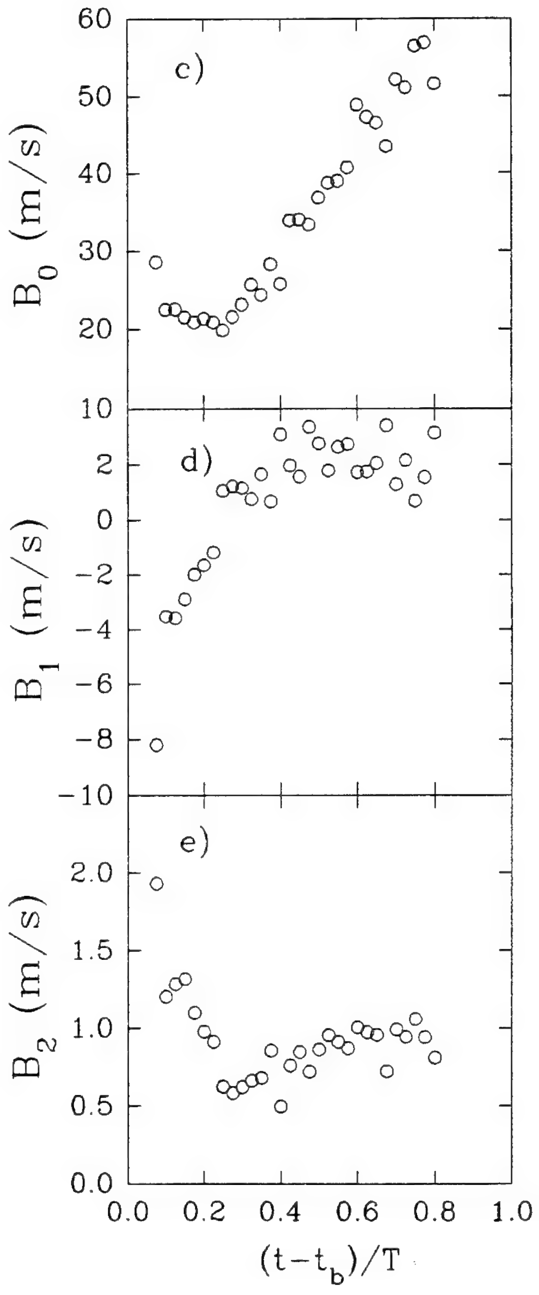


Figure 4.18: The fitting coefficients a) $A_0(t)$ and b) $A_1(t)$ for the exponential equation fitting the void-fraction profile as a function of radius (equation 4.8). Fitting coefficients c) Constant B_0, d) linear coefficient B_1 and e) second order coefficient B_2 as a function of time for the sound-speed profile fitted by equation 4.9.

4.8 Discussion and summary

We have presented measurements of the void-fraction field generated by 3D breaking waves from which we have computed several moments of the void-fraction field (figure 4.10 and 4.11). The void-fraction field evolved according to simple functions of time and showed decay rates consistent with the 2D experiments up to 0.5T after which the decay rates became slower. We suggest that the slower decay rates are caused by variances in the bubble plume evolution between the 2D and 3D plumes. As the 3D plume evolves, it entrains ambient fluid which makes the mean void-fraction decrease and the cross-sectional area of the plume increase. Clearly, the 3D plume can entrain fluid from the side whereas the 2D plume is constrained by the glass walls. Furthermore, the 3D plume has the potential to expand laterally and the 2D plume does not. It may be that this additional supply in ambient fluid coupled with a lateral expansion makes the 3D plume expand faster and to a larger size than in the 2D case.

The kinematics of the plume was investigated by tracking the centroid of the void-fraction field. In the 3D experiments where the secondary plume was found to be negligible, the horizontal centroid was found to move at 0.7C where C is the phase speed of the wave. This was consistent with the velocity of the primary bubble plume in the 2D experiment (refer to §3.3) and also consistent with recent field studies [Ding, 1992]. It was pointed out in the latter study that the velocity of breaking waves could be less than the velocity of the waves at the peak of the surface displacement spectrum (i.e. C) because breaking waves have frequencies greater than the spectral peak frequency. Finally, the horizontal centroid also showed that the bubble plume was advected horizontally by the orbital velocity of the wave after breaking.

The vertical centroid of the plume (figure 4.11b) in the 3D experiments was found to deepen at a speed of approximately 0.2H/T where H and T are the wave height at breaking and the wave period respectively. The maximum penetration depth ranged between 0.2H to 0.35H. These results were consistent with the 2D experiments for the most energetic breaking wave. The measurements of the vertical centroid of the bubble plume were scaled by H the wave height at breaking. It should be pointed out that H may not be the only scaling applicable for the penetration depth of the plume. Indeed, it would not be surprising if the penetration depth showed dependence on the wavelength, the wave packet slope [Rapp and Melville, 1990] and the rate of change of the geometrical properties of the wave (i.e. amplitude or slope).

The radial structure of the bubble plume was investigated based on the concept of an equivalent semicylindrical plume. It was found that the radial dependence of the void-fraction and sound-speed field can be described with relatively simple functions which depend on the bubble plume radius and time. These functional forms hold well for times up to 0.8T which covers the time period when the bubble plume is acoustically active [Loewen and Melville, 1993]. Newer models of bubble plume resonance are beginning to include radial variations of the void-fraction [Koller and Shankar, 1992b] distribution within the bubble plume. The measurements presented in this chapter should greatly help in determining if these variations are likely to yield resonance frequencies substantially different from those predicted using a uniform void-fraction across the entire bubble plume.

The radial structure of the sound-speed immediately after breaking ($t-t_b \leq 0.25T$), showed that the profile has a characteristic 'U' shape as shown in figure 4.19. As the forward face of the plunging wave impacts the water surface, a relatively 'smooth' cylinder of air is entrained under the water surface [Loewen, 1991]. As breaking evolves, the edges of this cylinder are first broken down into a bubbly mixture and this process progresses ultimately towards the center of the cylinder. Since the equation relating sound-speed to void-fraction [Wood, 1941] has a minimum sound-speed at 50% void-fraction, we can expect the sound-speed radial profile to have a 'U' shape (or a minimum at a finite radius) until the void-fraction in the center of the plume has dropped below 50% at which point the profile will be monotonically increasing from the center of the plume to the edges. This particular time would correspond to having the linear coefficient (B_1) of the sound-speed profile fit of figure 4.18d being equal to zero.

The acoustic results of Loewen and Melville [1993] have shown that the beginning of the low frequency (below 300Hz) pressure fluctuations generated by breaking is delayed by approximately 0.25T from the time at which the high frequency (above 1kHz) sound begins. The underwater video images of the bubble plumes showed no apparent peculiarities in the shape of the plumes or in the break up process of the plume which would explain this delay. For both the 2D (§3) and the 3D experiments it was observed that the beginning of the low frequency pressure fluctuations corresponded to the time at which the sound-speed profile became monotonically increasing. The reasons for this are still unclear but it may be that specific conditions on the shape of the sound-speed profile must be satisfied before collective oscillations can be initiated.

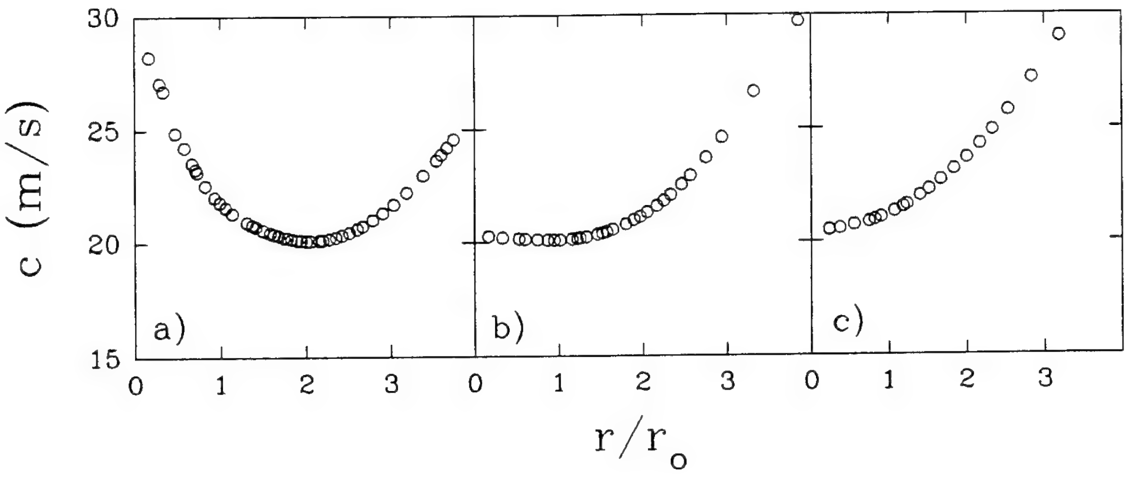


Figure 4.19: Sound-speed radial profile at a) t-t$_b$=0.075T, b) t-t$_b$=0.175T and c) t-t$_b$=0.275T for the wave packet amplitude with G=0.7.

Differences have been observed between fresh and sea water bubble population spectra [Monahan and Wang, 1992]. Recent "tipping bucket" laboratory experiments have shown order of magnitude differences in the number density of bubbles at their spectral peak [Monahan and Wang, 1992]. However, Monahan and Wang [1992] integrated their bubble spectra over 0.5sec windows to give the void-fraction as a function of time. They found that the maximum void-fraction levels in fresh and sea water did not differ substantially but that important differences were observed at later times when void-fraction levels in both fresh and salt water were significantly lower. In our experiments, we measured high void-fraction levels (0.3% to 100%) present in the first wave period following the onset of breaking. The findings of the tipping bucket experiments suggest that the results of our fresh water experiments would also apply in sea water for these short times after breaking.

Finally, the void-fraction measurements described in the chapter and in §3 along with independent measurements of the low frequency pressure fluctuations generated by breaking [Loewen and Melville, 1993] support the hypothesis that low frequency sound is generated by collective oscillations of the bubble plume. Figure 4.20 shows a comparisons between the measured volume oscillations (or collective oscillations) of the bubble plume based on acoustic measurements [Loewen and Melville, 1993] and the predicted frequency using the void-fraction measurements in this chapter and §3, and the model of Lu et al. [1990].

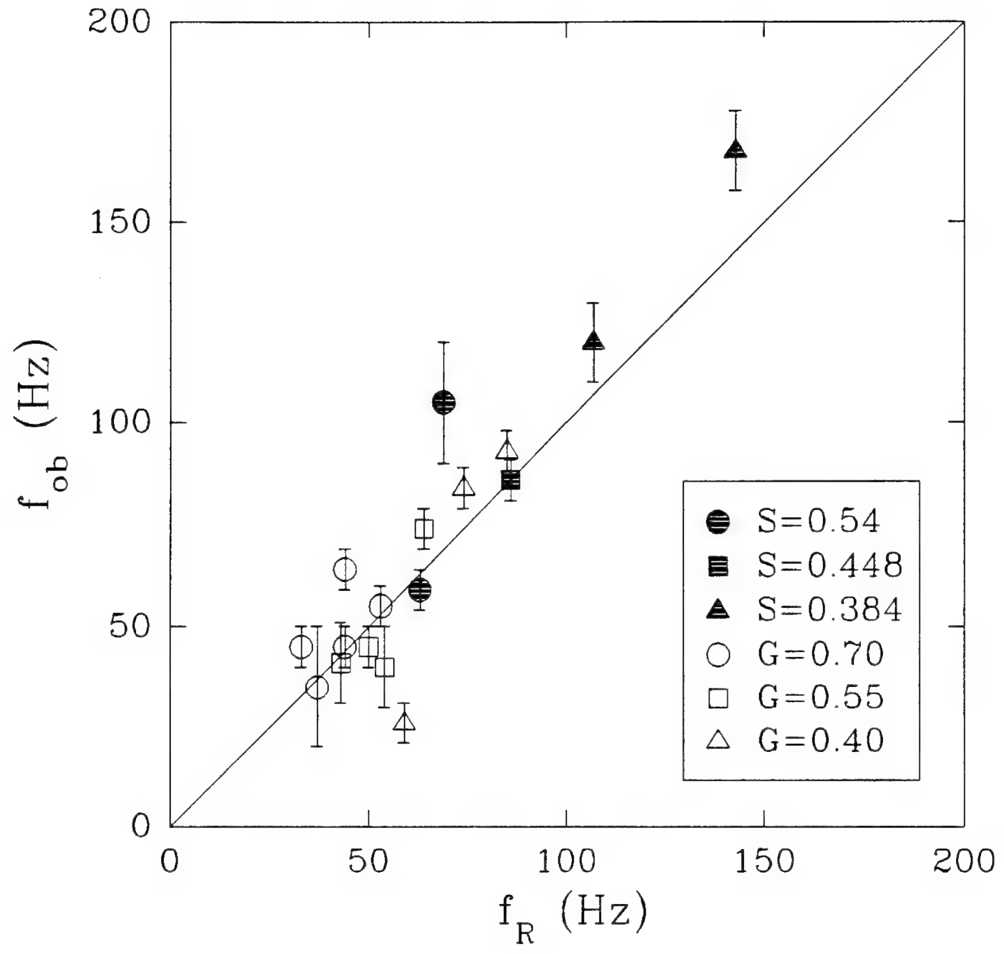


Figure 4.20: The observed dominant low frequency signal (f_{ob}) versus the computed resonance frequency f_R of a semicylindrical plume located at the free surface based on the void-fraction measurements of §3 (2D solid symbols) and this chapter (3D hollow symbols). The error bars indicate the 3dB width of the observed spectral peak. There are more than one point plotted for some events because comparisons were possible at more than one time in these cases. S refers to the slope of the wave packet in the 2D experiment and G refers to the gain of the wave packet in the 3D experiment. From Loewen and Melville [1993].

Chapter 5

Void-fraction measurements in the ocean

In this chapter, we report on the measurement of void-fractions in the field. Some of the work reported in this chapter first appeared in Lamarre and Melville [1992]. The experiments were part of the Surface Wave Dynamics Experiment (SWADE) conducted off the coast of Delaware [Weller et al., 1991]. We first describe the buoy and its instrumentation. Time-series from void-fraction probes deployed at three different depths are analyzed and correlated with wind speed and wave height measurements. Finally, subsurface video photography taken by a camera mounted on the buoy gives interesting qualitative information about the bubble plumes generated by breaking and the larger bubbles present within the plume.

5.1 The buoy and its instrumentation

A buoy was built to carry three void-fraction probes (refer to §2 for details on the field probe), one B&K model 8103 omnidirectional hydrophone, and a NEC TI-23A video camera (with a Fujinon model HF4.8B-SND4-1 auto-iris lens) mounted in an underwater housing fabricated by Underwater Video Vault (Spring, Texas). Figure 5.1 shows a line drawing of the buoy and its apparatus and Figure 5.2 shows the buoy on the deck of the Research Vessel Cape Henlopen from which it was deployed approximately 100km off the coast of Delaware during the last week of February 1991. The void-fraction probes were installed at depths of 20, 50 and 80cm from the water surface (the depth is measured from the center of the electrodes). The hydrophone was deployed at 1m below the surface and the video camera at 20cm. All signals were carried back to the ship by a set of five 100m cables which were supported along their length by small floats. The buoy was deployed upwind of the ship, and the ship was maneuvered to keep the tether slack

The electronics for the void-fraction probes were located on the ship. The signals from the probes were continuously sampled at 200Hz by a Metrabyte Das16 12bit data acquisition board mounted inside an HP Vectra 20MHz personal computer. The data was stored to hard disk every 5min. At the end of every day, the data on the hard disk were transferred to optical disks by an NHance 525WC optical drive.

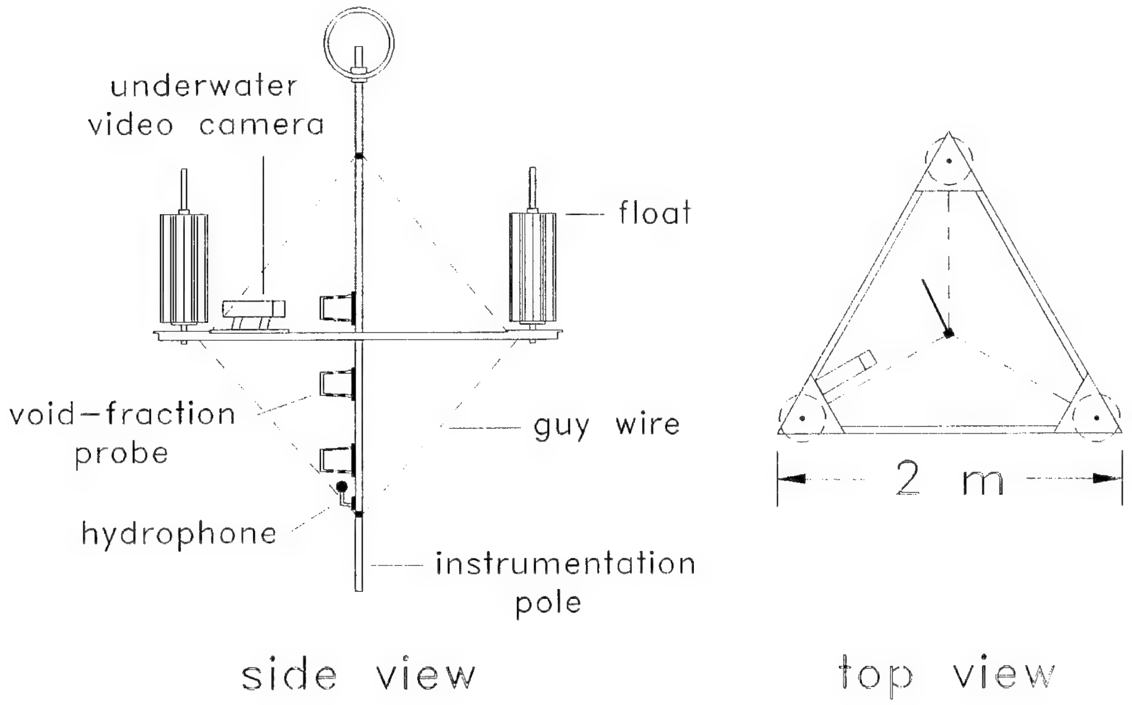


Figure 5.1: Line drawing of the buoy and associated instruments.

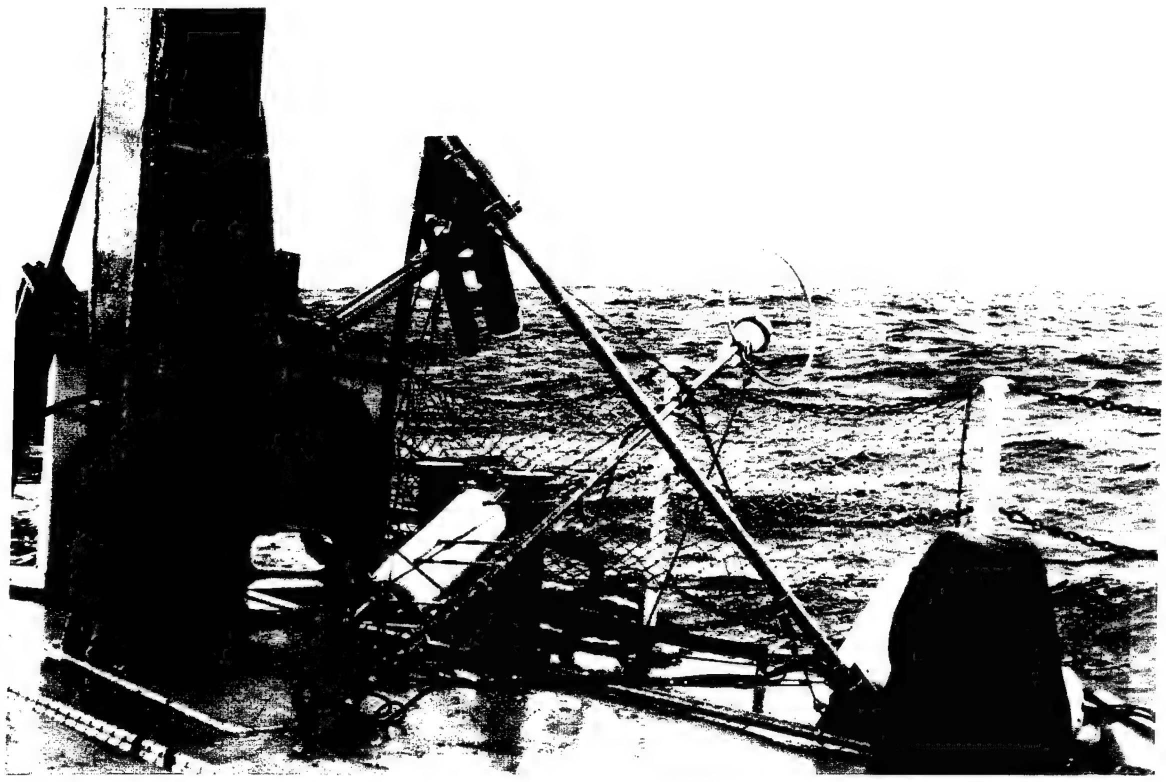


Figure 5.2: Photograph of the buoy and the mounted equipment on the deck of the R/V Cape Henlopen.

Hydrophone data were sampled and stored on a Panasonic SV-255 DAT recorder. Unfortunately, the hydrophone signal was not pre-amplified at the buoy before being carried back to the ship by the 100m coaxial cable. Because of this, the hydrophone data was heavily contaminated by electrical noise. Individual breaking events which were clearly visible on the video were not distinguishable on the audio track. The high-impedance output of piezo-electric transducers requires a pre-amplification stage to lower the impedance of the signal and hence to allow the signal to travel long distances without suffering from significant noise pick up.

Deployment #	Date	Time (EST)	Duration (min)	U_{10} (m/s)	SWH (m)	T_{dom} (s)	V.F. #	β (%)
1	02-25-91	9:00	55	14.1	2.7	8.3	20	5.0
		10:00	55	13.1	2.7	8.3	22	5.5
		11:00	45	13.4	2.8	6.2	33	7.6
2	02-26-91	9:00	60	4.5	1.7	8.3	10	2.3
		10:00	55	8.0	1.8	7.1	11	2.4
		11:00	40	6.6	1.7	8.3	5	1.7
3	02-27-91	12:00	35	14.9	2.4	6.7	9	2.9
		13:00	60	14.1	2.8	6.2	34	5.9
		14:00	60	13.1	2.7	6.7	30	5.6

Table 5.1: Three deployments during the SWADE field test. Wind speed, significant wave height (SWH) and dominant wave period (T_{dom}) are from nearby SWADE buoys located no further than 8km away. Wind speed was measured at 4m and estimated at 10m (U_{10}) using a logarithmic wind profile with a drag coefficient for neutral conditions [Large and Pond, 1981]. Wind speed was averaged over 8min, significant wave height and dominant wave period are 20min averages. The fraction of breaking waves per wave (β) is obtained from the number of void-fraction (V.F.) events above 1% divided by the duration multiplied by the dominant wave period. Eastern Standard Time (EST) is with respect to the center of the hourly record.

5.2 Void-fraction time-series

We have analyzed 465 minutes of data from a total of 560 minutes acquired during the three days of the field test. The wind and significant wave height ranged from 4.5m/s to 14.9m/s and 1.7m to 2.8m respectively. The wind was measured at 4m and estimated at 10m using a logarithmic profile with drag coefficient for neutral conditions [Large and Pond, 1981]. The wind and wave conditions along with the date and duration of the three deployments are reported in table 5.1. Each deployment is divided into hourly intervals to match the hourly data from SWADE buoys located no further than 8km away.

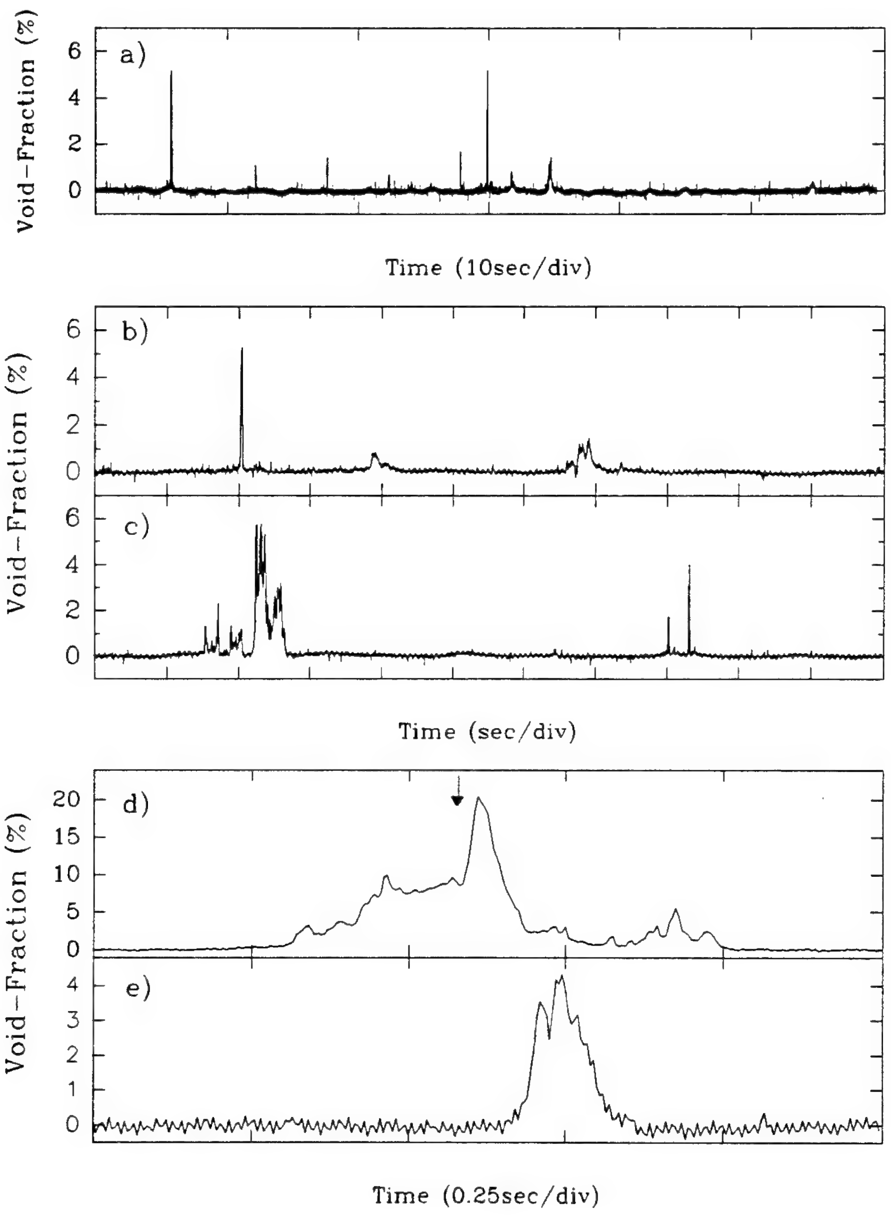


Figure 5.3: a) A one minute record of the void-fraction signal. b) Enlarged middle section of the time-series in a). c) Time-series with multiple events showing a wide range of signal amplitude and duration. d) Short time-series showing large void-fraction signal. The arrow corresponds to frame 8 of figure 5.7. e) Time-series for the gauge at 50cm. All other time-series are for the gauge at 20cm. Data sampled at 200Hz.

A one minute time-series extracted from a five minute record is shown on figure 5.3a for the gauge at 20cm. The record is from the third deployment at 14:00 hours (see table 5.1). Void-fraction events are found to be intermittent, with no events occurring in the remaining four minutes of the record (which is not shown here). Shorter time-series (11sec) from the gauge at 20cm (figure 5.3b,c) show details of the void-fraction signal for multiple events. The signal characteristically shows large variations in amplitude and duration. Large-scale void-fractions up to 24% are shown in the short time-series of figure 5.3d. The arrow on figure 5.3d corresponds to frame 8 of figure 5.7. Very few void-fraction events have been detected in the time-series at 50cm (10 for all three deployments) with the highest one being 5% (figure 5.3e). No events were detected at 80cm. In all of the data set, we have not detected simultaneous measurement of void-fraction events on separate gauges. This may be because of the limited data available at 50cm and the relatively large separation between the top and middle gauge. The very few void-fraction events observed at depth greater than 20cm indicates that the void-fraction field changes very rapidly with increases in depth.

We have processed all the time-series to give the signal duration and the maximum void-fraction for every event above a 0.5% threshold. An event is defined by a signal above the threshold and individual events are set apart when the signal returns below the threshold. The results are shown in the scatter plots of figure 5.4. Deployment 1 and 3 which had similar wind speed and significant wave height also show similar scatter plots (figure 5.4a and 5.4c). Significantly fewer occurrences were observed for the second deployment (figure 5.4b) where both the wind speed and the significant wave height were noticeably lower.

5.3 Void-fraction signal as a detector of breaking waves

The field measurements of the void-fraction can also be used to extract quantitative information on the statistics of breaking waves. Toba et al. [1971], and Holthuijsen and Herbers [1986] have quantified the fraction of breaking waves per wave (hereafter fraction of breaking waves) by using the presence of a whitecap at a fixed point on the surface. Similar measurements using the time derivative of a wave gauge signal have been made by Thorpe and Humphries [1980] and Longuet-Higgins and Smith [1983]. Holthuijsen and Herbers [1986] have reported on the significant differences in the above studies when correlating the fraction of breaking waves with wind speed. They have attributed the differences to the detection technique and its associated definition of breaking.

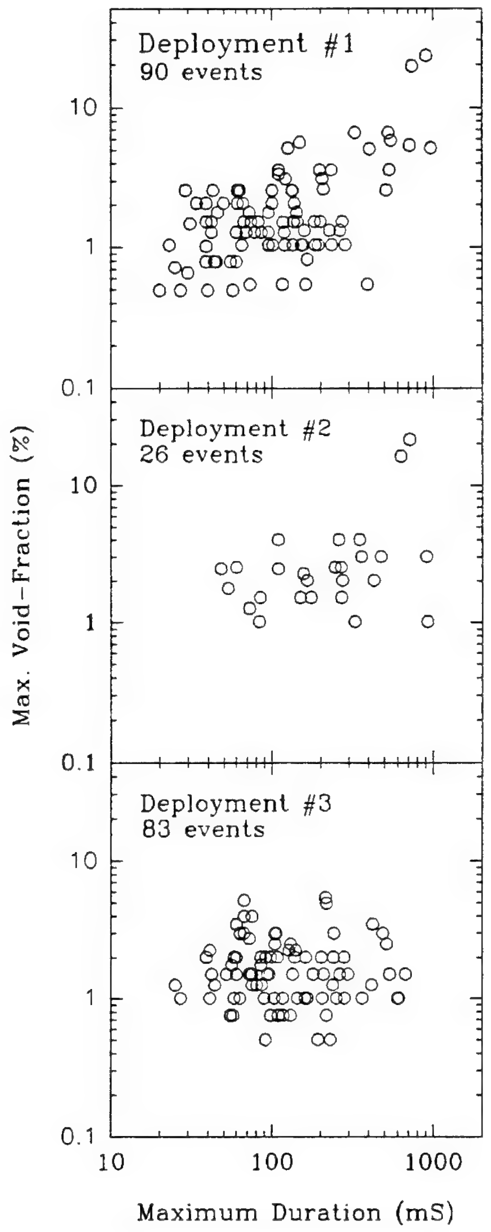


Figure 5.4: Scatter plots of event duration versus maximum void-fraction for a) deployment #1, b) deployment #2 and c) deployment #3. Each deployment has a duration of 155 minutes. The void-fraction is the maximum to occur for each event observed and the duration is the total time of the event above a threshold of 0.5% void-fraction. Note that one breaking wave may lead to multiple events (see figure 5.3c) so the duration should not be interpreted as the duration of a single breaking wave.

When used as a detector of breaking waves, the void-fraction probe should ideally be located very close to the surface. The minimum depth at which the probe can be deployed is restricted by surface effect (§2) and we can expect that there will always be some gently breaking waves which will not penetrate the minimum depth. Hence, the effect of the finite deployment depth should be to under-estimate the fraction of breaking waves.

On the other hand, in some time-series (figure 5.3c), we have observed multiple events which occur on a time scale much shorter than a typical wave period. These events are probably from the same breaking wave and therefore to use the void-fraction signal as a simple detection scheme of breaking may tend, in this case, to over-estimate the fraction of breaking waves.

The difficulty of discriminating between breaking and non-breaking events is a familiar problem in the study of breaking wave statistics [Holthuijsen and Herbers, 1986] both from an instrumentation and from a definition point of view. The preliminary aim here is to present the void-fraction probe as a possible tool for the study of breaking wave statistics.

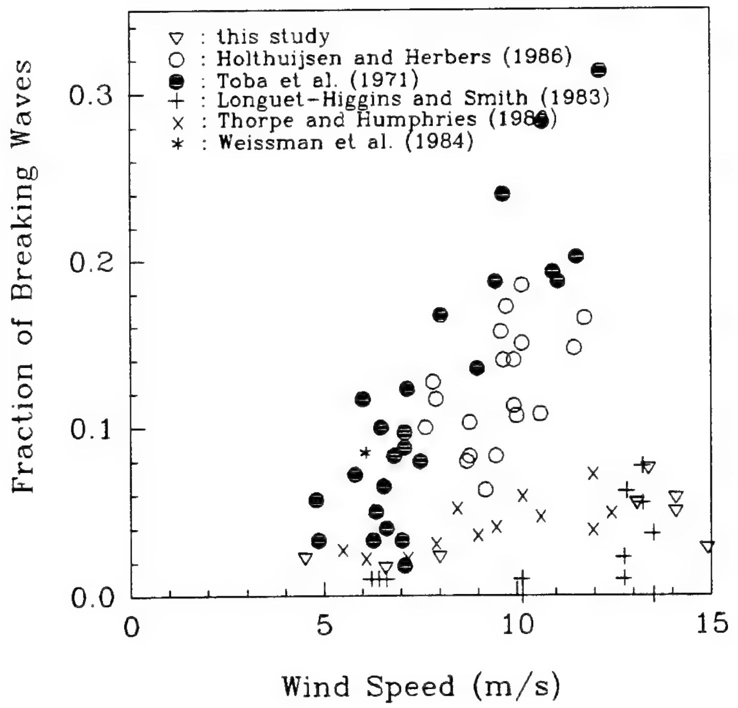


Figure 5.5: Fraction of breaking waves per wave as a function of wind speed at 10m. Adapted from Holthuijsen and Herbers [1986].

With this simple detection device, the fraction of breaking waves is given by the number of void-fraction events per second multiplied by the dominant wave period. We now define a void-fraction event as one which has its maximum signal above a 1% threshold; but we point out that the results do not differ significantly for lower thresholds down to 0.5%. The number of such events for each deployment is reported in table 5.1. Figure 5.5 shows the fraction of breaking waves as a function of wind speed. Our data shows scatter comparable to previous studies and is consistent with results based on the time derivative of a wave gauge signal [Thorpe and Humphries, 1980; Longuet-Higgins and Smith, 1983]. It may be that our results are more compatible with the above two studies because the time derivative technique tends to detect events associated with rapid changes in the surface elevation which may lead to a greater penetration of the bubble plume.

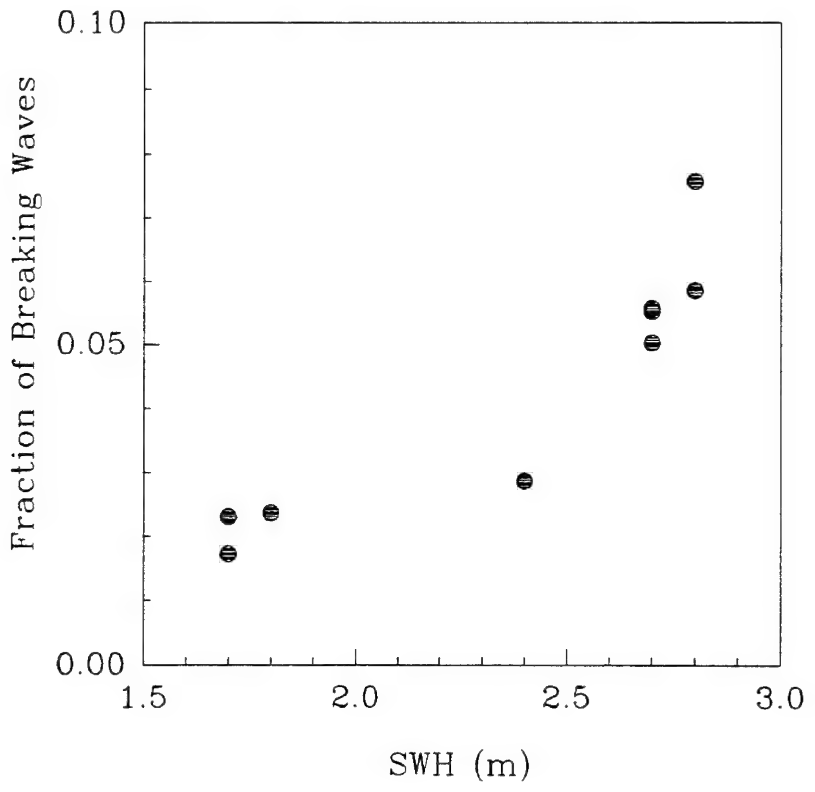


Figure 5.6: Fraction of breaking waves per wave as a function of significant wave height.

Figure 5.6 shows the fraction of breaking waves as a function of significant wave height. There is clearly an increase with significant wave height but the relatively small data set limits the significance of the correlation. Attempts to correlate the fraction of breaking waves with wind speed and significant wave height for higher thresholds of the void-fraction signal (3% and 5%) were less successful because of the limited number of events at these higher thresholds. We found no significant changes in the results of figure 5.5 and figure 5.6 for lower thresholds down to 0.5% void-fraction.

5.4 Video photography

Wave breaking conditions were sufficiently strong during the experiment to provide interesting video images of occasional large-scale bubble plumes generated by breaking waves. The underwater video photography provided impressive visualization of the formation and evolution of the bubble plume.

The subsurface video camera looked directly horizontally at the void-fraction probe located at a depth of 20cm. The housing was equipped with an 8cm diameter Plexiglass dome port as the water interface lens. The dome port creates a virtual image of the real object being observed. This image is located 2cm to 20cm in front of the dome depending on the distance of the real object which can range from the dome to infinity. The camera was mounted with a lens having a 4.8mm focal length and a measured 18cm depth of field. Therefore, by focussing the camera at 11cm everything in front of the dome port appeared in focus.

Figure 5.7 displays photographs taken from the underwater video tape. They were generated by taking photographs (Kodak Plus-X pan 125 film at a 1/15sec shutter speed and 5.6 aperture) of a 38cm Panasonic WV-5470 video monitor with the recorder set on still mode. The photographs clearly show the birth and evolution of a bubble cloud. The section of the frame's horizontal member shown at the bottom of the photographs measures approximately 1.2m and is located 1.1m in front of the camera at a depth of 0.4m. The series of photographs in figure 5.7 corresponds to a time-series from the upper void-fraction probe shown on figure 5.3d (the arrow on figure 5.3d corresponds to frame 8 of figure 5.7).

The video also permitted the observation of large bubbles. Their minimum dimensions were inferred from the known diameter of the dome port (8cm). Figure 5.7 (frames 5, 6, and 7) shows large bubbles from the breaking event with the largest for this event being 5mm in radius (frame 7). In almost every situation when bubbles reached the front of the dome port, it was possible to identify bubbles with radii in the millimeter range.

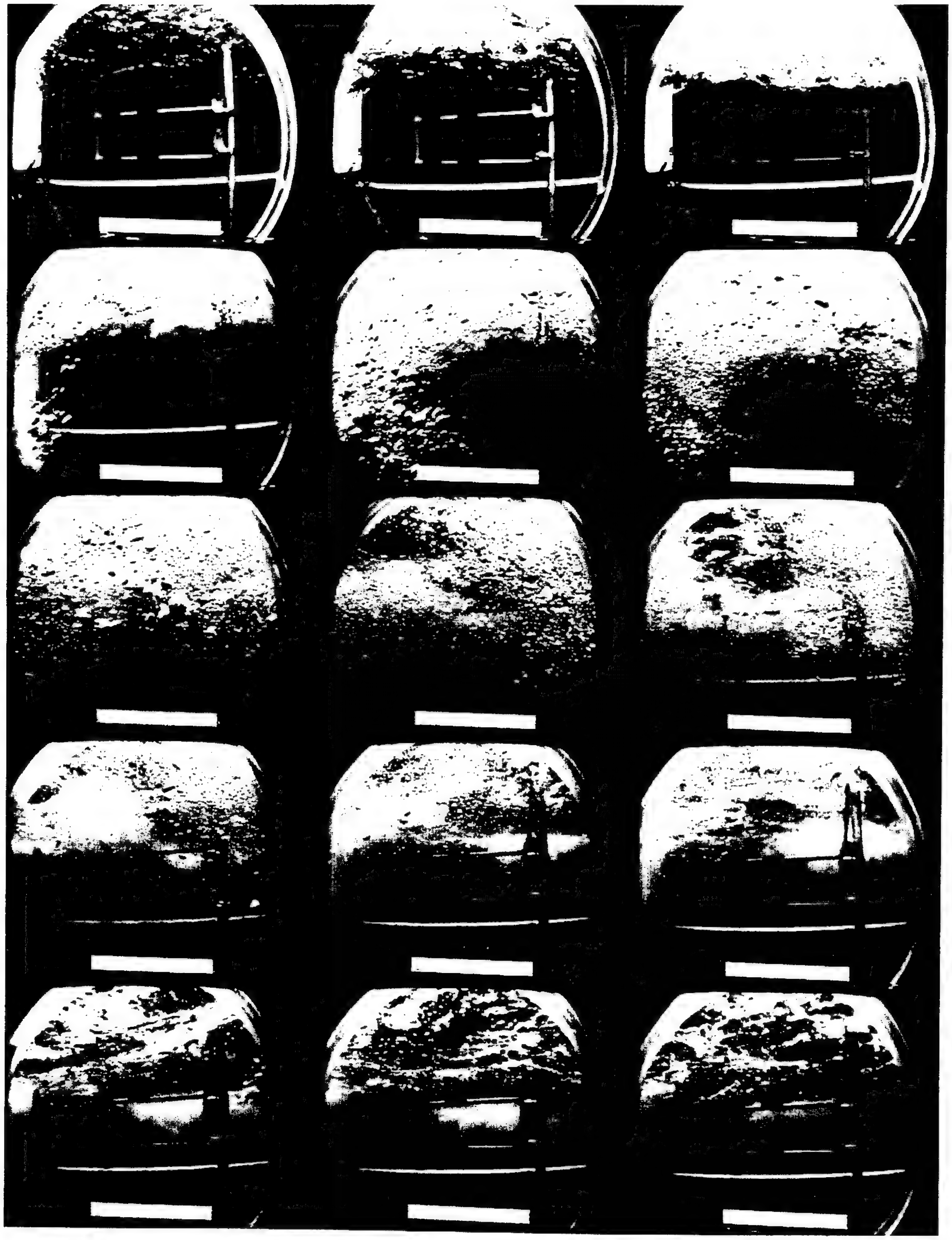


Figure 5.7: (Prevous page) Photographs of the bubble plume taken from the video recording. The photographs evolve in time from left to right and from top to bottom. Frame 1 to 7 are 0.133s apart, frame 7 to 12 are 0.267s apart and frame 12 to 15 are 0.533s apart. Frame 8 corresponds to the arrow on figure 5.3d. The largest bubble (5mm radius) in this sequence is seen in frame 7.

5.5 Discussion and summary

In this chapter, we reported on field measurements of void-fraction within the first meter of the ocean surface.

Void-fraction time-series at 20cm and 50cm depth have shown occasional high void-fraction signals up to 24% which are consistent with laboratory measurements (§3 and §4) and several orders of magnitude larger than time-averaged values previously reported [Walsh and Mulhearn, 1987]. An underwater video camera has confirmed that these high void-fraction levels were from bubble plumes generated by breaking immediately above the sensor.

Void-fraction events above a threshold of 0.5% were found to be sporadic at a depth of 20cm and no occurrences have been detected at 80cm for the three deployments reported here. This pointed out to the fact that high void-fraction levels (above 0.3%) decrease very rapidly within the first meter below the ocean surface. It was found that the number of high void-fraction events at 20cm depth increased with increasing sea state but the duration of the events and their maximum void-fraction did not significantly change with sea state.

Preliminary measurements showed that the fraction of breaking waves per wave, based on the detection of breaking by the void-fraction gauge, was dependent on significant wave height and wind speed. The dependence on wind speed was compared with data of previous investigators and it was found to be consistent with their lower bound data.

The principal shortcomings of the present detection technique were identified. In order to generate a signal on the void-fraction probe, breaking waves must entrain air bubbles at a finite depth. Hence, small breaking waves may not be detected by the void-fraction probe if the bubbles they entrain do not reach the probe. Furthermore, the detection is very sensitive to the measuring volume of the probe. For example, a few bubbles entrained in a very small measuring volume could generate a detectable void-fraction signal but the same bubbles could remain in the background noise of the instrument if the measuring volume was large. The problem with differentiating multiple events which are triggered by the same breaking waves was also pointed out. The above problems could be minimized by deploying an horizontal array of void-fraction probes with smaller measuring volume. The smaller measuring volume would allow these probes to make measurements very close to the surface (O(cm)). With these changes, the void-fraction probe could become a sophisticated instrument for the study of breaking wave statistics.

Experiments recently reported by Su et al. [1993] on the measurement of void-fraction at sea for depths comparable to the ones reported in this study showed results consistent with the present work. They measured high void-fractions (up to 60%) at a depth of 25cm and a wind speed of 15m/s and they reported that the void-fraction events above their instrument detection threshold of 0.2% were very rare at depths greater than 50cm which is in agreement with the present study.

They have also reported on the detection of breaking waves with their void-fraction sensor. They have found that the fraction of breaking waves per wave increased with a decrease in the selected void-fraction threshold. Hence, their results were consistent with the upper bound data shown in figure 5.5 when they selected a small threshold (0.2% void-fraction) and they were consistent with the lower bound data when they selected a high-threshold (5 to 10% void-fraction). At a void-fraction threshold of 1%, a depth of 25cm and a wind speed of 15m/s, they observed the fraction of breaking waves per wave to be 0.26 while for the same void-fraction threshold and wind speed and a depth of 20cm the present study reports ~0.06. It is believed that these differences are mainly due to the different measuring volumes of the two instruments. We would expect that for a smaller measuring volume more void-fraction events would be recorded. For example, Su et al. [1993] used a probe with a measuring volume of ~17cm^3 while in the present study we have used a measuring volume of ~300cm^3 (because of fringing of the electric field, it is difficult to estimate accurately the true size of the measuring volumes, see §2.4.3). The high sensitivity of the void-fraction measurements to the size of the measuring volume suggests caution in comparing different void-fraction studies obtained with different probes.

Video photography has shown the evolution of episodic bubble plumes generated by breaking waves. In these videos, we have also observed frequent occurrences of large bubbles (at least 6mm radius) in the field. In the past, bubbles of this size were not believed to exist in the ocean [Lu et al, 1990]. Unfortunately, no research has been conducted on making quantitative measurements of the transient large bubble population (say 0.5mm to 5mm) in "active" plumes generated by breaking. Large bubbles have a sufficiently long residence time in the water to transfer gas [Broecker and Siems, 1984] and to radiate sound generated at their creation. Hence, quantitative measurements of these large transient bubbles could lead to a better understanding of their contribution to the low-frequency ambient sound (say 0.6 to 6.0kHz) and to global air-sea gas transfer.

SECTION II

MEASUREMENTS OF SOUND-SPEED

Chapter 6

Instrumentation for the measurement of sound-speed near the ocean surface

This chapter deals with the development of a field instrument for the measurement of sound-speed near the ocean surface for depths of 50cm and greater. The technique described in this chapter was developed for the measurement of sound-speed in bubbly mixtures for acoustic waves of low frequencies ~5kHz, but it is also applicable for higher frequencies.

We first give an overview of the ocean buoy and the instrumentation developed for these measurements. We examine the necessary tradeoffs for acoustic measurements at low frequencies which are performed close to the water surface. We move on to look in detail at the design of the hydrophones' support module. Particular attention is given to acoustic interference from the support itself, the water surface and other structural buoy elements. We then look in detail at the various subsystems of the sound-speed measuring instrumentation: conditioning of transmit and receive signals, real-time data acquisition and data processing. Finally, we present results from controlled tests in the laboratory and in a lake. The results confirm the validity of the technique and establish some basic performance criterion. Data from the field which demonstrate the operation of the instrument in an ocean environment are presented in §7.

6.1 General description of the buoy and its instrumentation

A light buoy was built to carry all relevant probes and instrumentation (figure 6.1). The buoy was equipped with six acoustic sound-speed measuring modules. Each module included a transmit and a receive omnidirectional hydrophone both rigidly supported on a stainless steel rod. The transmit hydrophone is an ITC-1042 and the receive hydrophone is an ITC-1089E. Refer to appendix G for detailed specifications on probes and instruments described in this chapter.

Other instrumentation on the buoy included a Sea-Bird SBE 3 temperature sensor, three void-fraction probes (refer to §2 for specifications), a NEC TI-23A video camera mounted in a Video Vault (Spring, Texas) underwater housing, an ITC-8181A omnidirectional hydrophone with built in pre-amplifier for sampling ambient sound and an 88kHz Furuno model FCV-561 upward looking sonar. The sonar was used to image the bubble clouds as has been previously done by Thorpe and Hall [1982] and Farmer and Vagle [1989].

The total weight of the buoy in air was approximately 45kg. This weight was supported in water by 14 football-shaped floats each with a buoyancy of 4.5kg. From our video measurements, we found that the buoy tracked the water surface to within ± 10cm.

Two 150m underwater multi-conductor cables served as links between the buoy and the instrumentation on board the ship. The cables were connected to two underwater enclosures mounted on the buoy. The enclosures served as junction boxes between the buoy probes and the multi-conductor cables. The buoy was tethered from a ship and the ship was maneuvered to keep the tether slack at all time.

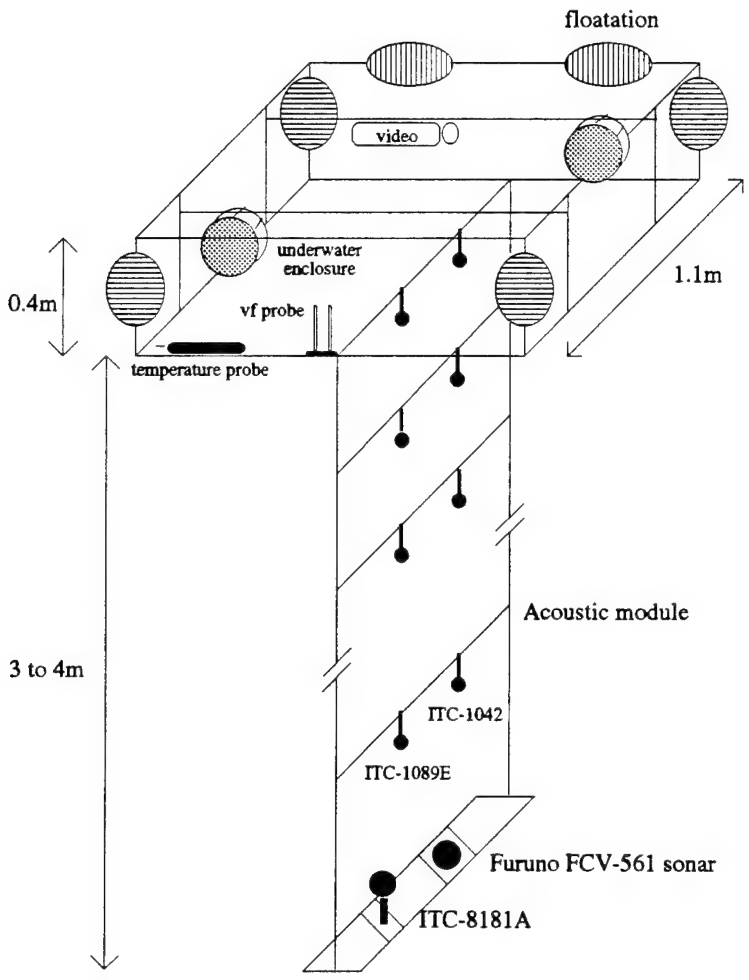


Figure 6.1: Sketch of the buoy and its instrumentation.

6.2 Balancing acoustic frequency, pulse length, separation of hydrophones and depth

The sound-speed measurement technique described in this chapter is based on estimating the travel time (or time-delay) taken by an acoustic pulse as it propagates from a transmitter to a receiver. The time-delay is computed by cross-correlating the transmit and received pulses and by locating the peak in the cross-correlation function. Since the distance between the hydrophones is constant, and known from measurements in bubble free water, one can readily compute the speed of sound in the bubbly mixture from the knowledge of the time-delay. It is critical, with this technique, to eliminate or at least minimize any reflections of the acoustic pulse by nearby objects or by the water surface which would bias the location of the peak in the cross-correlation function.

The elimination of unwanted reflections requires balancing four parameters: 1) the acoustic pulse frequency f (or more precisely the wavelength $\lambda = Cf$ where C is the speed of sound in water or in the bubbly mixture), 2) the pulse length L_p, 3) the separation distance between the transmit and receive hydrophone L_d (i.e. the direct path) and 4) the depth of the hydrophones. The depth of the hydrophones is important because in our application the reflection from the surface is the dominant and often the closest reflective path. Note that the pulse length $L_p = N\lambda$ where N is the number of periods. Balancing these parameters requires that

$$L_p + L_d < L_r \qquad (6.1)$$

where L_r is the length of the closest reflective path. In essence, this equation says that reflections from nearby objects or from the water surface must arrive at the receive hydrophone after the entire pulse from the direct path has reached the receiver. The determination of each of the above three lengths requires careful compromises which will greatly influence the characteristic and performance of the instrument.

Due to their limited bandwidth, underwater transducers are restricted in their ability to transmit short sound pulses. For spherical hydrophones, two to three periods is about the minimum that can be achieved. This therefore sets a lower bound on N.

We can readily place an upper limit on the acoustic frequency. Medwin et al. [1975] have estimated, from their field measurements of the speed of sound at sea, that below 25kHz the sound-speed is no longer dispersive. Their measurements were performed at depths greater than 3m. For shallower depths, where the bubble density is expected to be greater, this 25kHz cutoff frequency could be lower. (Recall that the dispersive nature of underwater sound at higher frequencies is caused by bubbles which introduce important phase shifts when excited by acoustic waves at or near their resonance frequencies. Appendix B gives a detailed discussion of these effects). It will be shown in §7 that 20kHz was a more reasonable estimate for the cutoff frequency of dispersion (for wind speeds ~8m/s).

The lower frequency limit is dictated by the proximity of the surface and by the ability of the transducer to transmit sound with high enough acoustic intensity to dominate the ambient oceanic sound. For a small 2-3cm diameter transducer, this lower limit is ~3-6kHz. Lower frequencies can be achieved with larger transducers but their size makes them difficult to deploy and they are significantly costlier.

One of the novelties of this study was the measurement of sound-speed as close to the surface as possible. Because of the large difference in acoustic impedance between water and air, we expected strong reflections from the surface. Assuming that the transmit and receive hydrophones are positioned at the same depth d (i.e. the pulse propagates horizontally), the length of the reflective path from the surface is given by

$$L_r = \sqrt{d^2 + \left(\frac{L_d}{2}\right)^2}. \qquad 6.2$$

Using equation 6.1 and 6.2 and assuming $L_p \cong 2.0\lambda$ we can construct a table of L_d as a function of hydrophone depth d and acoustic frequency f (see table 6.1)

		d			
		0.25m	0.5m	0.75m	1.0m
f	2.5kHz (λ=0.6m)	-	-	0.3m	1.05m
	5.0kHz (λ=0.3m)	-	0.55m	1.55m	3.0m
	10.0kHz (λ=0.15m)	0.25m	1.5m	3.6m	6.5m

Table 6.1: Maximum possible length of the direct path as a function of acoustic frequency and hydrophones depth. The transmit and receive hydrophones are assumed to be at the same depth. The length of the acoustic pulse $L_p=2\lambda$ where λ is the acoustic wavelength.

The length of the direct path L_d must be kept short enough to permit measurements as close to the surface as possible and long enough to obtain a meaningful spatial averaging. The later was difficult to estimate considering that no such measurements have been reported in the past.

We selected a direct path of length L_d=40cm and a transmit acoustic frequency of 5kHz. Of course, any frequency above 5kHz can also be used. The number of periods in the pulse was set at 2 to 3, the minimum achievable. Table 6.1 gives d=0.5m as being the smallest achievable depth with the parameters selected above.

6.3 The support module of the hydrophones

The support module which holds the transmit and receive hydrophones is shown in figure 6.2. The bodies of the hydrophones are held by thin sleeves which are welded to 6mm diameter vertical rods which in turn are welded to the main supporting rod 13mm diameter. All members are made of stainless steel which provided minimal member cross-sectional dimensions for a given rigidity. The rigidity is important since the hydrophones must be kept at a fixed distance from one another. The small cross-sectional dimensions of the members minimized their acoustic reflections. Extensive acoustic testing of this support module is presented in subsequent sections where it will be shown that it generated negligible acoustic interference. The distance between the hydrophones is L_d=40cm. The support rod is attached on both sides to the buoy's main vertical members (see figure 6.1). The length of the reflective path to these vertical members is L_r=110cm which is long enough to delay the acoustic reflections from the buoy's vertical members.

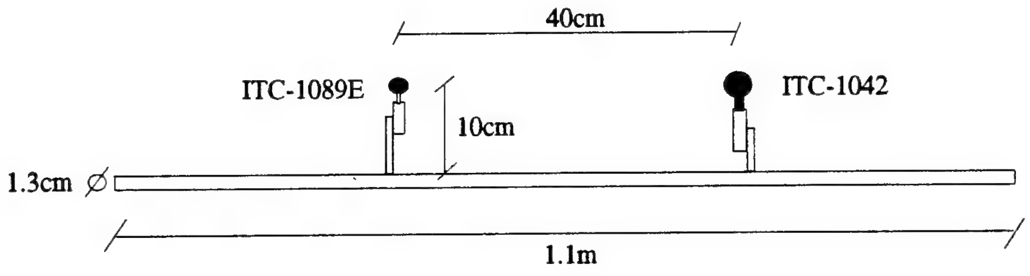


Figure 6.2: Sketch of the support module.

6.4 Transmit and receive acoustic systems

The transmit and receive acoustic systems are shown in figure 6.3 and figure 6.4. The boxes with a double line around them refer to expansion cards mounted inside a PC

ALR-Veisa 386-33MHz computer. The computer was equipped with an expansion chassis from Industrial Computer Source model NODE/XTC/ATC-D to accommodate the large number of expansion cards (a total of 7).

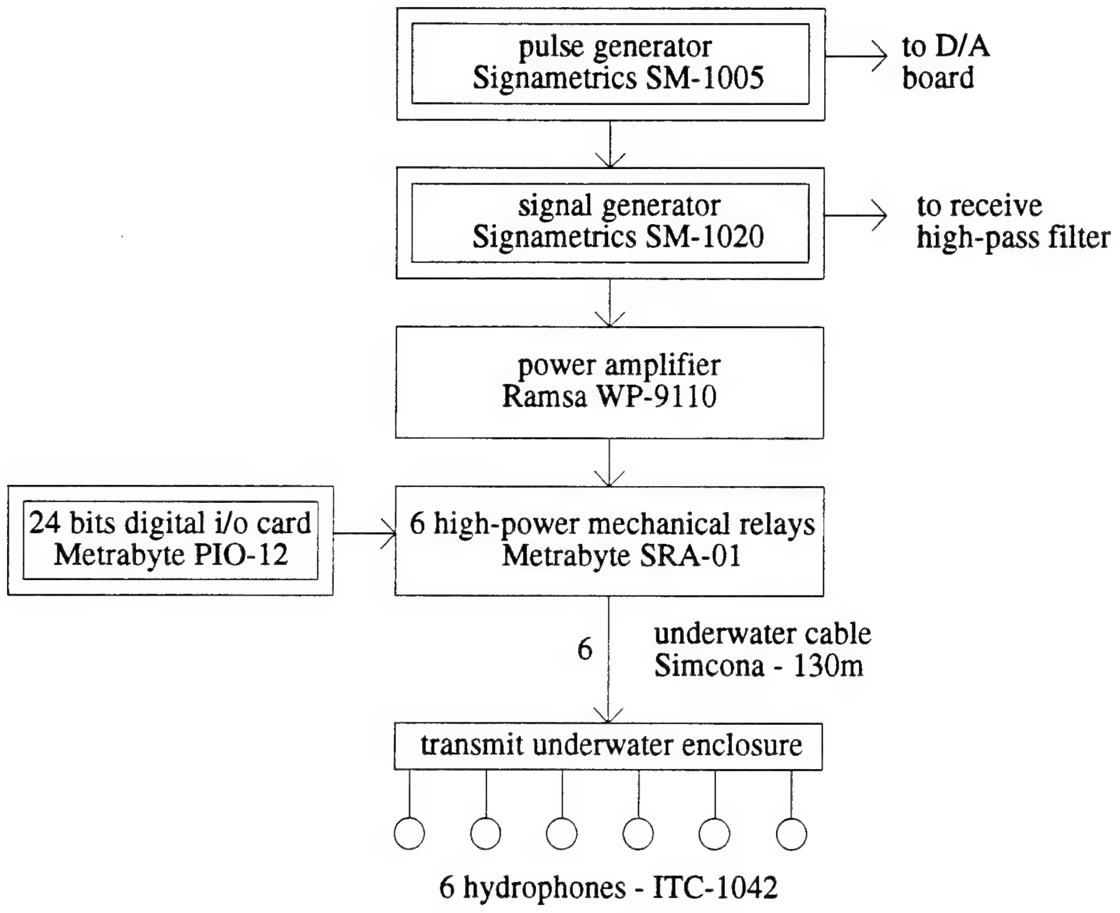


Figure 6.3: Bloc diagram of transmit acoustic system.

Both the transmit and receive systems are synchronized by a pulse generator which acts as the main control clock. The pulse generator triggers the signal generator which in turn transmits the waveform recorded in its RAM to a power amplifier. The amplitude of the signal transmitted by the signal generator was computer controlled. This has been a valuable feature which helped in offsetting the well known 12dB/octave transmit response of spherical hydrophones. Hence, as the signal frequency increased, the amplitude of the transmit signal output by the signal generator was reduced. Table 6.2 shows the various gain settings used as a function of frequency. The San Diego and Woods Hole experiment refer to two field experiments that were conducted in the fall of 1992. More details about these experiments are given in §7.

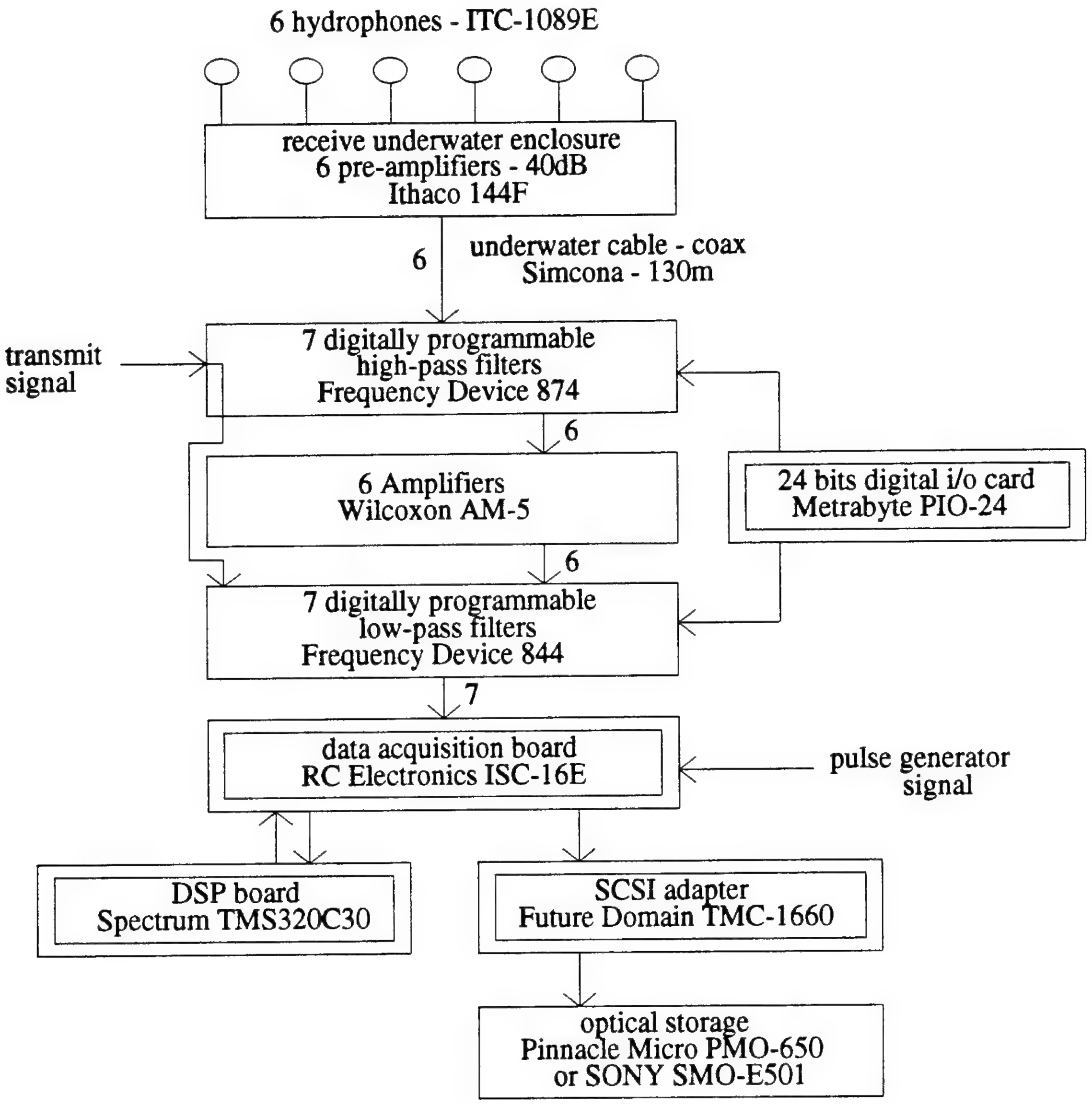


Figure 6.4: Bloc diagram of receive acoustic system.

After the power amplifier stage, the pulse is routed to the appropriate transmit hydrophone by a bank of high-power mechanical relays which are controlled by a digital I/O card. The signal travels from the ship to the transmit underwater enclosure located on the buoy through an underwater cable made of 15 individually shielded paired conductors. The underwater enclosure acts as a junction box between the six hydrophones and the long underwater cable.

San Diego experiment

f (kHz)	signal generator gain	power amplifier gain
5.0	5.0	27dB
7.5	3.5	27dB
10.0	2.0	27dB
15.0	1.0	27dB
20.0	0.6	27dB

Woods Hole experiment

f (kHz)	signal generator gain	power amplifier gain
5	7.0	27dB
10	3.0	27dB
composite	5.0	27dB

Table 6.2: Signal generator and power amplifier gain settings for the two field experiments described in §7. The voltage gain of the signal generator is applied to the transmit signal of maximum amplitude of unity (i.e. a gain of 5 means that the signal generator will output a pulse of 5V peak amplitude.) The composite pulse was composed of 4 distinct frequencies. It is described in detail in §7.

The transmitted pulse is received by a hydrophone located 40cm horizontally away from the transmit hydrophone (see figure 6.2). The received signal is first amplified by a 40dB pre-amplifier located in the receive underwater enclosure on the buoy (figure 6.4). After this first amplification stage, the signal is carried back to the ship via an underwater cable made of RG-174 coaxial cables. The use of coaxial cables for carrying acoustic signals over long distances was a prudent, if not necessary, measure. The use of separate enclosures and cables for carrying the signals too and from the ship was necessary in order to reduce electrical cross-talk of the transmit pulse on the receive signal.

At the ship, the receive acoustic pulse is passed through a high-pass filter, a second 40dB amplification stage and a low-pass filter. The signal must be conditioned in that order. After the pre-amplification stage performed at the buoy, the signal contains high-amplitude low-frequency components generated by the hydrostatic pressure of surface waves. (Note that the high-pass f_{3dB} response of the pre-amplifier is 0.5Hz.) At the ship, the signal must therefore be high-pass filtered first to eliminate these low-frequency hydrodynamic fluctuations. The second stage of amplification enhances the weaker high-frequency acoustic signal and the final low-pass filtering stage reduces the noise for frequencies above the transmit acoustic frequency.

The filters essentially produce tight band-pass filtering around the transmit acoustic frequency. Table 6.3 gives the various amplification gains and filter cutoffs for the various acoustic frequencies used. The cutoff frequencies of the filters were digitally programmable from the computer with the help of a digital I/O card. Notice that the transmit pulse is also passed through the filter bank in order to match the inherent time delays introduced by filtering. Finally, the conditioned signals were sampled by a 1MHz data acquisition board. Real-time processing of the transmit and received pulses was performed by a dedicated DSP board and the results along with the raw data were stored on 600Mb optical disks. The details of the data acquisition and data processing technique are explained in subsequent sections.

San Diego experiment

f (kHz)	pre-amplifier gain (dB)	high-pass filter cutoff (kHz)	amplifier gain (dB)	low-pass filter cutoff (kHz)
5.0	40	0.50	40	7.5
7.5	40	0.75	40	15.0
10.0	40	1.00	40	13.2
15.0	40	1.50	40	19.2
20.0	40	2.00	40	24.2

Woods Hole experiment

f (kHz)	pre-amplifier gain (dB)	high-pass filter cutoff (kHz)	amplifier gain (dB)	low-pass filter cutoff (kHz)
5	40	1.0	40	6.4
10	40	1.0	40	13.2
composite	40	0.5	40	48.0

Table 6.3: Gain of pre-amplifiers and amplifiers, and cutoff frequencies of the high-pass and low-pass filters for various acoustic frequencies. The settings for the two field experiments described in §7 are shown. The composite pulse was composed of 4 distinct frequencies. It is described in detail in §7.

6.5 Time-series and spectra of the acoustic pulse.

The transmit signal was designed by windowing a sinusoidal function. A Bartlett window was selected because it offers a good compromise between spectral peak width and sidelobe fall-off. The window was adjusted such that approximately two periods of the sinusoid was transmitted. A longer window would provide a narrower spectral peak which is desirable in order to minimize the spectral energy at bubble resonance frequencies. On the other hand, a longer window also increases the length of the pulse L_p and this reduces our ability to make measurements close to the water surface. Figure 6.5a shows a time-series of the transmit signal used for all frequencies between 5kHz and 20kHz. The time axis is normalized by the acoustic wave period T. Figure 6.5b shows the spectra of the signal where the frequency axis as been normalized by $f_o=1/T$. The second harmonic is -24dB down from the main peak. Combining this roll off with the characterisitic +12dB/octave of the spherical transducers give an effective -12dB/octave prior to band-pass filtering.

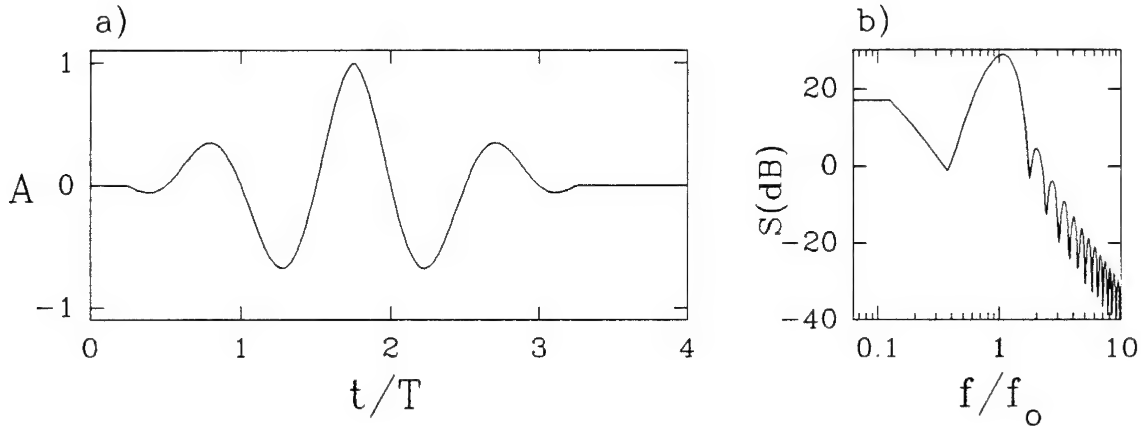


Figure 6.5: a) time-series of the pulse transmitted to the power amplifier. b) Spectrum of time-series in a). The time axis in a) and frequency axis in b) are normalized by the period T and the frequency f_o of the acoustic wave.

The data acquisition system (described in the next section) samples both the transmit and the receive signal. As shown in figure 6.4, both the transmit signal and the receive signal are passed through the high-pass and low-pass filters. Figure 6.6a and figure 6.7a show typical field data of transmit (above) and receive (below) acoustic pulses at 5kHz and 10kHz respectively. The transmit and receive hydrophones were both located at a depth of 0.5m. The direct acoustic path and the reflection from the surface are clearly visible in both figures. Also notice that for a smaller transmit signal, the receive signal is much stronger for the 10kHz pulse (i.e. figure 6.7a). This is due to the transmitting response of spherical hydrophones which typically rise at 12dB/octave up to the resonance frequency of the transducer (~80kHz).

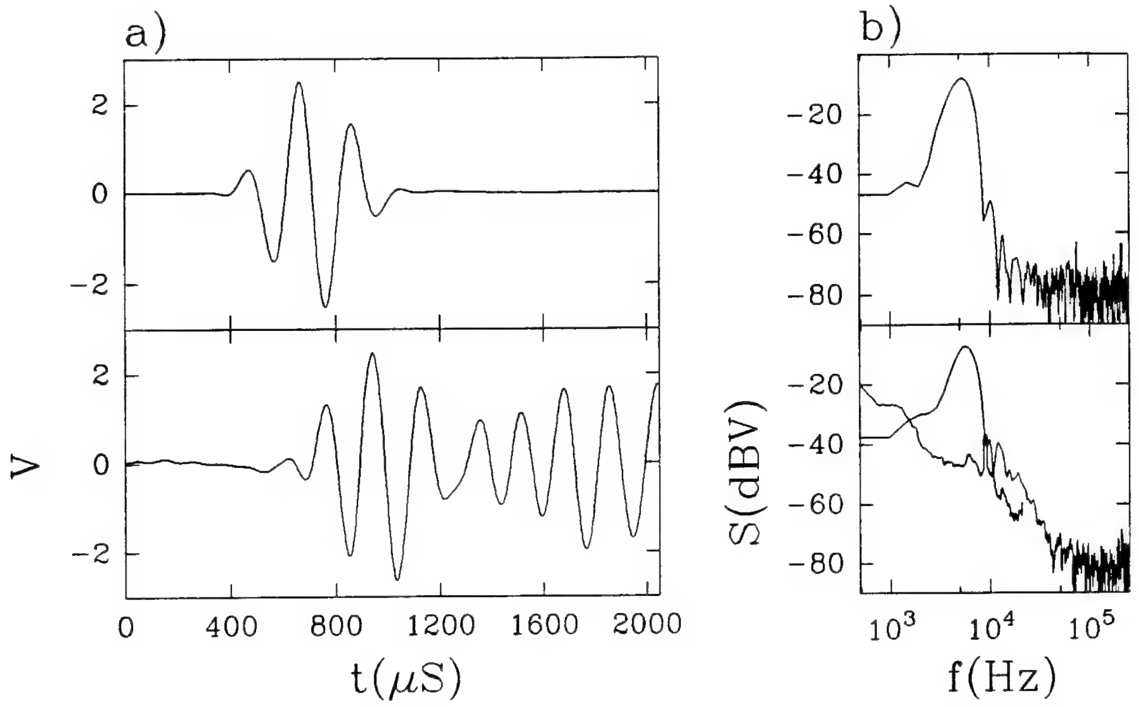


Figure 6.6: a) time-series of transmit (above) and receive (below) 5kHz acoustic pulses. b) Spectra of time-series in a). Note that the receive time-series was windowed with a 500 point Bartlett window in order to eliminate the reflections from contributing to the spectrum. The second line on the lower spectra is the ambient sound for W_s = 8m/s.

Figure 6.6b and figure 6.7b show the spectra for the time-series shown in figure 6.6a and figure 6.7a. The receive time-series was windowed to eliminate the reflective path from contributing to the spectrum. Clearly, the spectra roll off very rapidly at higher frequencies. This is a desirable feature since we wish to make sound-speed measurements at frequencies well below bubble resonance. Hence, the spectral energy at higher frequencies is kept to a minimum. The second lower line on the receive spectra of figure 6.6b and figure 6.7b represent the ambient noise level for wind speeds around 8m/s. The pulses are therefore 30dB to 40dB above the ambient noise level which would give a signal to noise ratio of 50 to 100.

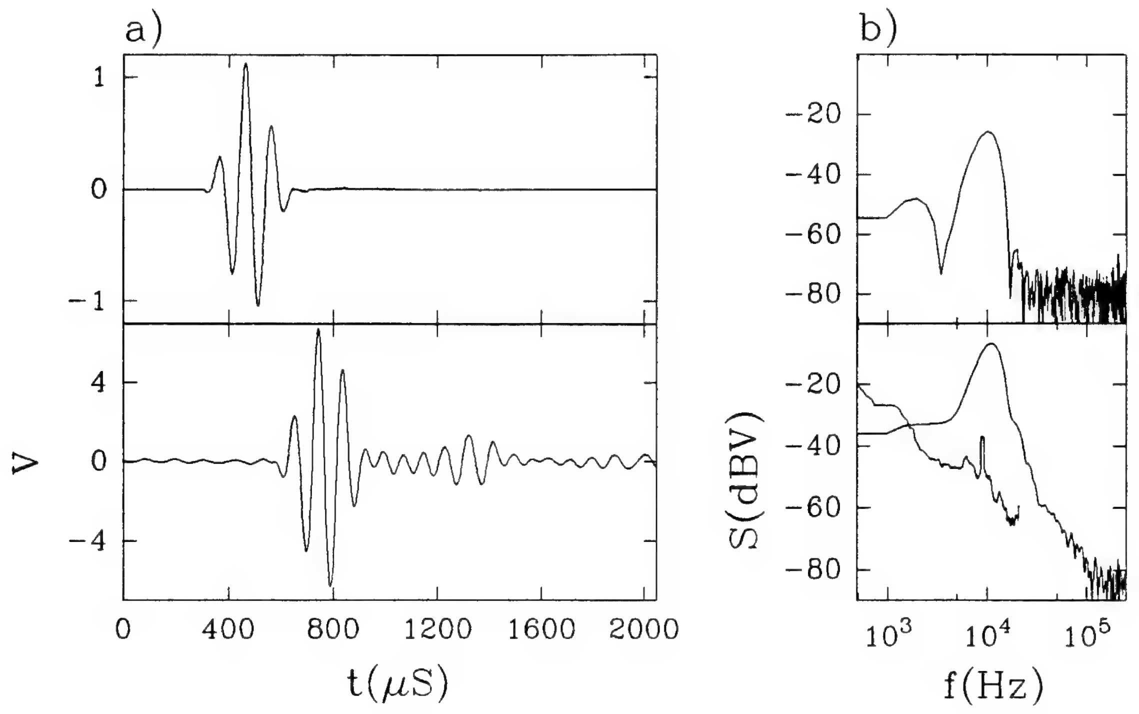


Figure 6.7: a) time-series of transmit (above) and receive (below) 10kHz acoustic pulses. b) Spectrum of time-series in a). Note that the receive time-series was windowed with a 250 point Bartlett window in order to eliminate the reflections from contributing to the spectrum. The second line on the lower spectra is the ambient sound for $W_s = 8$m/s.

6.6 Data acquisition

In this section and the next, we describe in detail the real-time data acquisition and data processing systems. The ability to do real-time monitoring of the sound-speed fluctuations was a very valuable feature, especially during the field experiments. It permitted detecting problems immediately and it gave the opportunity to investigate various alternatives for the sampling rate and the depth of the hydrophones. The source code for the data acquisition and processing software is given in appendix I.

The data acquisition system is essentially made up of three major bloc components as shown in table 6.4. The primary bloc deals with initialization of various boards. First, the frequency of the pulse generator is set (refer to figure 6.3). The signal generator, which transmits the acoustic pulse, is then enabled. Next, the cross-correlation code is uploaded to the DSP board and the program is reset and put on standby. Finally, the real-time graphic windows are initialized and also put on standby.

Initialization

- initialization of transmit system
- initialization of DSP board
- initialization of real real-time graphics

Real-time data acquisition and data processing loop

- selection of transmit hydrophone
- initialization of data acquisition board
- selection of receive hydrophone
- data acquisition board waits for trigger
- after reception of trigger, sampling proceeds
- data is written to RAM
- data is uploaded to DSP board
- computation of time-delay on DSP board
- cross-correlation result downloaded from DSP
- result displayed on screen

Data acquisition of support sensor and data storage

- sampling of temperature probe
- read video time-code generator
- transfer data from RAM to optical disk
- write cross-correlation results to optical disk

Table 6.4: The three principle bloc components of the data acquisition system.

The second bloc constitutes the core of the system. All instructions in the bloc must be performed in order to complete one measurement of the speed of sound at a specific depth. First, one of the six transmit hydrophones is selected by a 24 bit digital I/O card (of which only 8 bits are used) which enables a bank of high power mechanical relays (refer to figure 6.3).

Next, the RC Electronics 12 bit data acquisition board is initialized for the proper sample rate, channel selection and trigger mode. The channel selection defines which receiving hydrophone is selected. The selected transmit and receive hydrophones are always located on the same support module. The data acquisition is then placed on hold as it waits for the trigger from the pulse generator. After reception of the trigger, the transmit and receive pulses are sampled at 500kHz each for a duration of 2048μS. This sample window is long enough to accommodate large reductions in sound-speed down to the ~300m/s. After sampling is completed, the two 1024 data arrays are downloaded from the data acquisition board and stored to a 4Mb RAM disk. The RAM disk was used as a temporary fast storage device before the final transfer to optical disk.

The transmit and receive data arrays are then uploaded to the DSP board for computation of time-delay. This particular step is detailed in the next section. Upon completion, the result (i.e. time-delay) is downloaded back to the PC where it is displayed by a real-time graphics routine. It should be pointed out that the calculations on the DSP board were performed in parallel with other PC functions. This is possible because the DSP board has its own processor. It can therefore operate as an independent device but the handshaking, or data transfer, between the DSP board and the PC must be synchronized by software running on both processors.

The operations of the second bloc are repeated for each new sound-speed measurement. The peak performance of the system permitted sampling of all six transmit-receive modules at 4Hz. This gave a total of 24 sound-speed measurements per second. At that rate, it took 40sec to fill the 4Mb RAM disk with raw data. At sea, the system was modified to only save the raw data on every other sample. Hence, measurements were taken continuously for 80 seconds before the buffer was full.

The third and final bloc in figure 6.4 samples the water temperature and reads the video time-code generator (refer to appendix G for details on these devices). Also, the computed time-delays and the data stored to RAM (nearly 4Mb) are finally transferred to optical disk. Two different kind of optical drives were used during the field work. The slower Sony drive stored the 4Mb buffer in 28 seconds and the faster Pinnacle Micro drive accomplished this task in 10 seconds.

6.7 Digital signal processing

The calculation of the time-delay between the transmit and the receive acoustic pulses is accomplished by locating the position of the peak of the cross-correlation function. It is well known that for arrays greater than 64 points, the cross-correlation can be computed significantly faster by using the Fast Fourier Transform (FFT) than by computing it using the shift-multiply-add method (known as the direct method).

With the FFT method, the forward FFTs of the two arrays to be cross-correlated are computed first. The resulting complex arrays are multiplied to obtain the cross-spectrum which is finally run through an inverse FFT to obtain the real cross-correlation array. The location of the peak in the cross-correlation array immediately gives the time-delay.

The computationally intensive parts of this calculation are the two forward and the inverse FFTs. For example, on a 33MHz PC it will take approximately 100mS to compute a 1024 FFT and approximately three times as much to compute a cross-correlation . Adding to this the overhead time for data acquisition, storage and display gives ~300mS for one measurement of the sound-speed. Clearly, this was too slow to appropriately sample at several depths the sound-speed field which we suspected to have fluctuations at surface wave frequencies (~2Hz and smaller).

A Spectrum TMS320C30 Digital Signal Processing (DSP) board was used to speed up the calculation of the time-delay and hence permit real-time data acquisition and data processing. The cross-correlation code was written is C and it used a highly optimized forward and inverse FFT assembly routines. The DSP board gave a factor of 5 improvement over the PC performance for cross-correlation computations. This included all the overhead time required to pass the data from the PC to the DSP board. Furthermore, the ability of the DSP board to run in parallel with the PC processor gave a further factor of 2 increase in performance.

6.7.1 Interpolation of the cross-correlation

A sampling rate of 500kHz per channel gives a 2μS resolution on the location of the peak of the cross-correlation. Considering that the location of the peak will be around 267μS (i.e. 0.4m path divided by 1500m/s) this gives a resolution of 0.7% (or 10m/s) on the determination of the speed of sound.

We can further increase this resolution by making use of the very high signal to noise ratio of the receive pulse (SNR~10^2) and its narrow bandwidth. These two properties of the transmit and receive signal give a very smooth cross-correlation function. For example, the cross-correlation function for the transmit and receive pulses of figure 6.6a and figure 6.7a are shown in figure 6.8a and figure 6.9a. Since these functions are very smooth, it is possible to interpolate them and consequently increase the resolution of the exact position of the cross-correlation peak.

At present, nine points located around the peak of the cross-correlation function are used in a cubic spline routine to perform the interpolation (refer to figure 6.8b and figure 6.9b). The interpolation is performed on a grid 20 times finer than the original. The computation of the interpolation is very fast and it adds negligible amount of time over the actual computation time of the cross-correlation. Figure 6.8c and figure 6.9c show an enlarged version of the cross-correlation peak. Clearly, the interpolation has significantly increased the resolution of the peak in figure 6.8c but it has done only a marginal improvement on figure 6.9c. Of course, this is due to the distribution of the data samples on the original time-series.

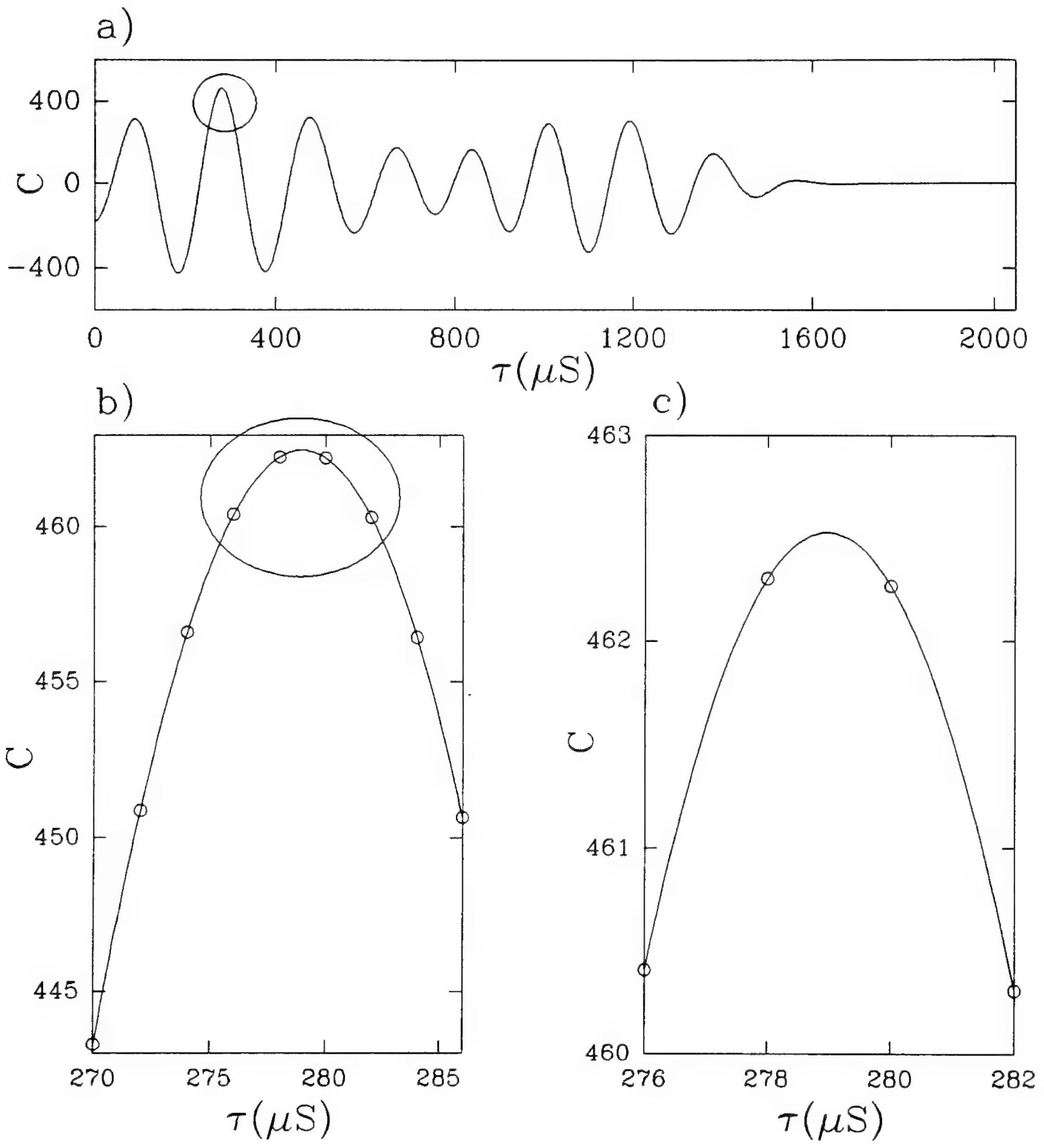


Figure 6.8: a) Cross-correlation function for the time-series in figure 6.6a (i.e. 5kHz pulses). b) Expanded section around the peak of a). c) Expanded section around the peak of b). The data from the cross-correlation are shown as (o) in b) and c). The solid line in b) and c) is the interpolation.

The laboratory tests, discussed subsequently in this chapter, show that this interpolation technique has given an O(10) increase in the resolution of the sound-speed measurements in laboratory conditions. Appendix D and E give a detailed analysis of this interpolation process and some insight on its sensitivity to noise, sampling rate and the number of points in the spline fit.

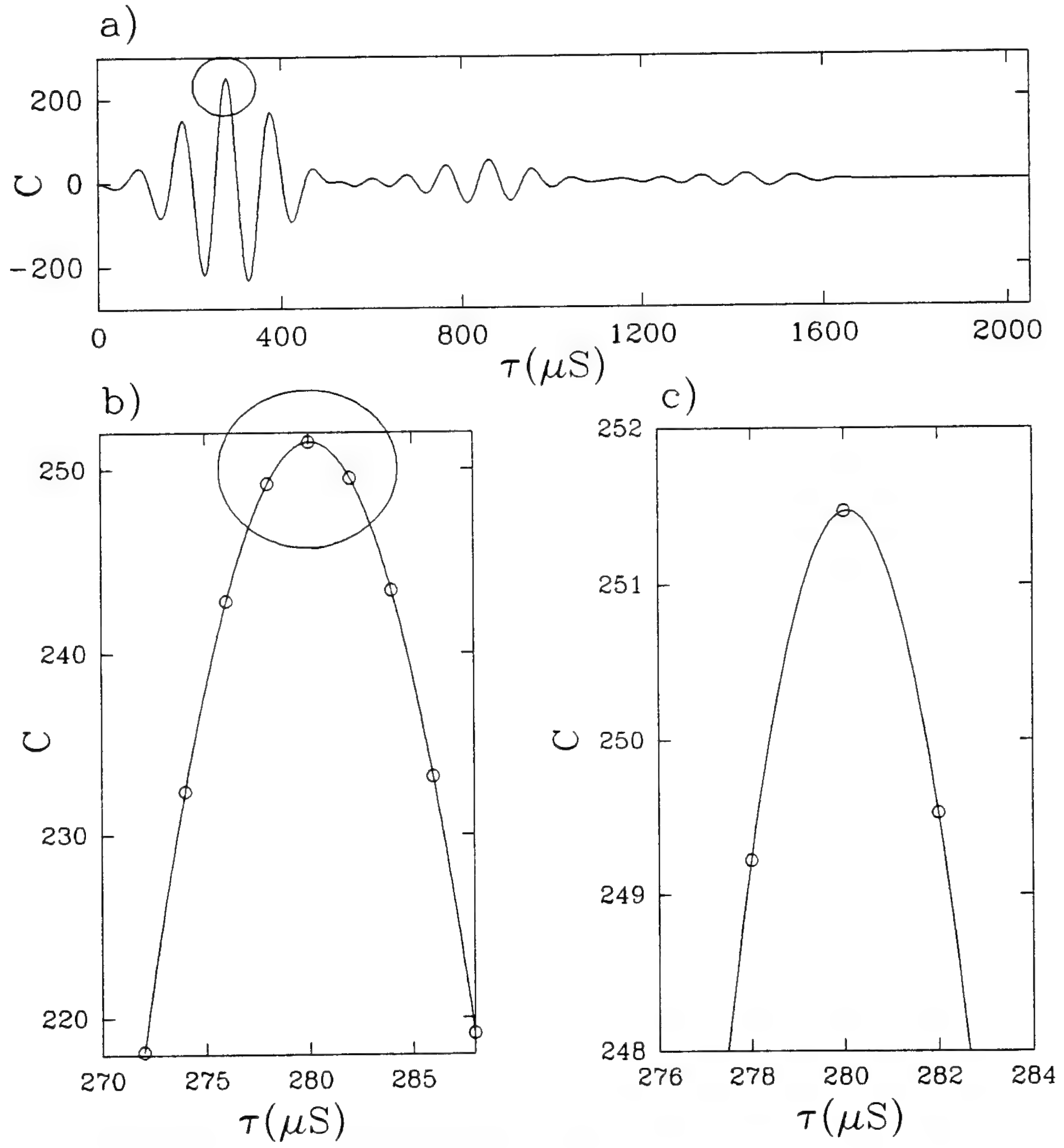


Figure 6.9: a) Cross-correlation function for the time-series in figure 6.7a (i.e. 10kHz pulses). b) Expanded section around the peak of a). c) Expanded section around the peak of b). The data from the cross-correlation are shown as (o) in b) and c). The solid line in b) and c) is the interpolation.

6.8 The response time of the transmit hydrophone

In the receive bloc diagram of figure 6.4, both the transmitted and the received signals are high-pass and low-pass filtered before being sampled. In essence, this procedure eliminates the lag caused by the filters from the computed time-delay. However, there are still other lags such as the hydrophone's response time and the amplifiers' response time which are not accounted for. Because of their high-bandwidths, we expect the amplifiers to introduce negligible delays (see appendix G for bandwidths specifications). The delay caused by the response of the transmit hydrophone is more difficult to calculate or measure and thus, a few scaling arguments can help in quantifying the magnitude of its effect on the sound-speed measurements.

The measured time-delay can be written as

$$\tau_m = \tau_t + \tau_d \tag{6.3}$$

where τ_t is the true propagation time of the pulse over a distance d separating the source and the receiver and τ_d is a constant time-delay caused by the response of the hydrophone and any other electronic devices. Actual measurements of τ_m over a separation of 40cm in a tank of known water temperature and thus known sound-speed (approximately 1500m/s) give $\tau_m \cong 267\mu s$. Consequently, we can assert that τ_d is much smaller than τ_t since for a propagation path of 40cm the true propagation time-delay for a sound-speed of 1500m/s would be $\tau_t = 266.6\mu s$. The separation distance can be written as

$$d = c(\tau_m - \tau_d) \tag{6.4}$$

where c is the speed of sound. Now assume that c is composed of a constant part and a fluctuating part $c = c_w + \Delta c$ and Δc is the signal we are interested in measuring. Substituting into equation 6.4 and rearranging we obtain

$$c_w + \Delta c = \left(\frac{d - c_w\,\tau_d}{\tau_m}\right) - \frac{\tau_d}{\tau_m}\Delta c \tag{6.5}$$

where the numerator of the term in brackets is constant and can be replaced by an effective distance d_e. Thus, equation 6.5 becomes

$$\Delta c = \left(\frac{d_e}{\tau_m} - c_w\right)\left(1 - \frac{\tau_d}{\tau_m}\right)^{-1} \cong \left(\frac{d_e}{\tau_m} - c_w\right)\left(1 + \frac{\tau_d}{\tau_m}\right). \tag{6.6}$$

Clearly, the last term in the last brackets is much smaller than unity since $\tau_d << \tau_m$ and thus the effect of the hydrophone's response time τ_d is negligible (a conservative estimate would be $\tau_d \leq 0.01\tau_m$). Consequently, equation 6.6 becomes

$$\Delta c \cong \frac{d_e}{\tau_m} - c_w. \qquad 6.7$$

One can determine the value of the effective distance d_e in bubble-free water where the sound-speed $c = \Delta c + c_w$ is know. In the presence of bubbles, the time-delay τ_m is obtained from the cross-correlation and it is used, along with the prior knowledge of d_e, to infer the speed of sound in the bubbly mixture. Note that c_w is obtained from temperature measurements in both bubbly and bubble-free water.

6.9 Acoustic interference tests

A series of tests have been performed in the laboratory and in a small lake to ensure that the hydrophone support module did not interfere with the acoustic pulses and that, under controlled sound-speed variations, the instrument could accurately track the changes to the speed of sound.

6.9.1 Acoustic interference from a solid rod

Prior to the design of the acoustic module, a preliminary study was conducted to investigate the acoustic interference caused by the supporting rod. A solid rod was positioned in parallel and close to the direct path between the transmit and the receive hydrophones. A schematic of the tank and of the position of the hydrophones is shown in figure 6.10a. The tank dimensions are 3m by 1.5m by 1.0m in depth. The hydrophones were supported by two 13mm diameter aluminum rods and positioned 0.5m below the surface and 0.4m apart.

The position of the rod for the interference tests is shown in figure 6.10b. A solid stainless steel rod 13mm is positioned adjacent to the hydrophones. The distance d from the head of the hydrophones is varied between 2cm and 8cm. A short pulse is transmitted and the received signal is compared with the case when no rod is present.

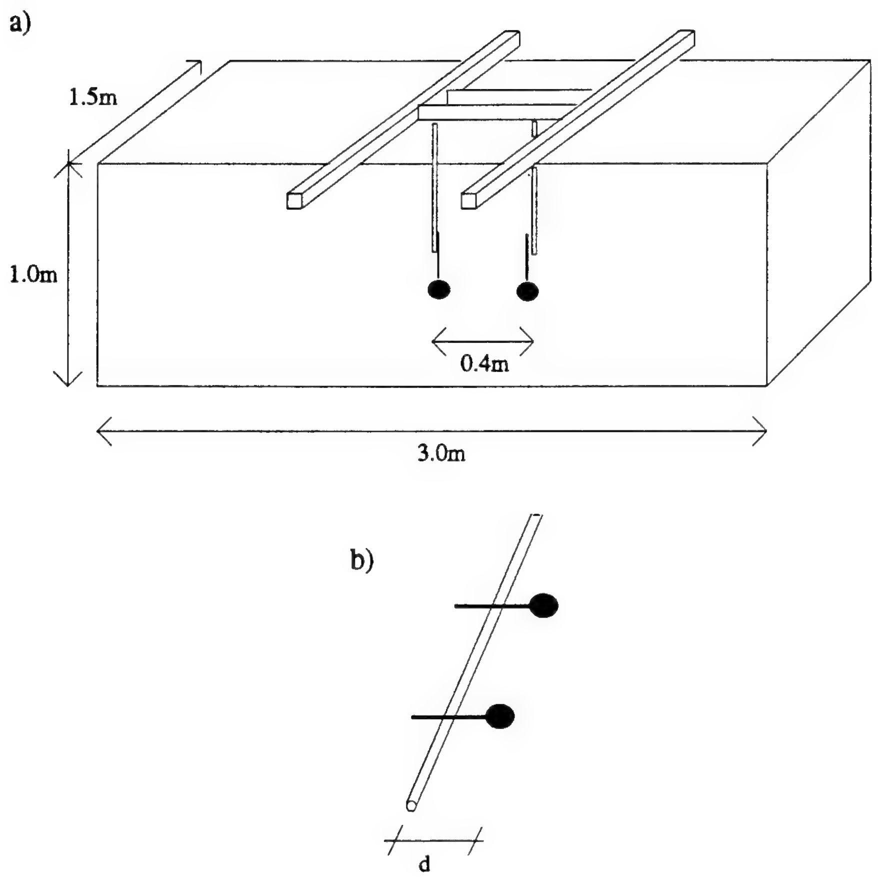


Figure 6.10: a) Sketch of the laboratory tank and of the position of the hydrophones for the interference test. b) Sketch defining the position of the rod for the interference test.

Figure 6.11 shows results for 10, 25 and 50 kHz pulses. At 10 kHz (figure 6.11a) we can observe slight amplitude differences for d=2cm and essentially no phase differences for both d=2cm and d=8cm when compared to the 'no rod' case. Recall that the phase is critical in measuring time-delay and it would be important that the rod supporting the hydrophones introduced no phase errors and ideally minimal amplitude errors although amplitude changes are less critical. At 25kHz and 50kHz (figure 6.11b,c) the phase differences become more significant especially for d=2cm. Notice the earlier arrival of the acoustic pulse which occurs in the presence of the solid rod. This phenomenon was not investigated in detail but it is believed that it is caused by acoustic waves which are coupled to the structure and propagate at a speed greater than the speed of sound in the fluid. At 25kHz (figure 6.11b) the second pulse arriving at 700μs is the reflection from the surface. This pulse has less phase distortion because the propagation direction is nearly perpendicular to the solid rod as opposed to parallel for the direct path.

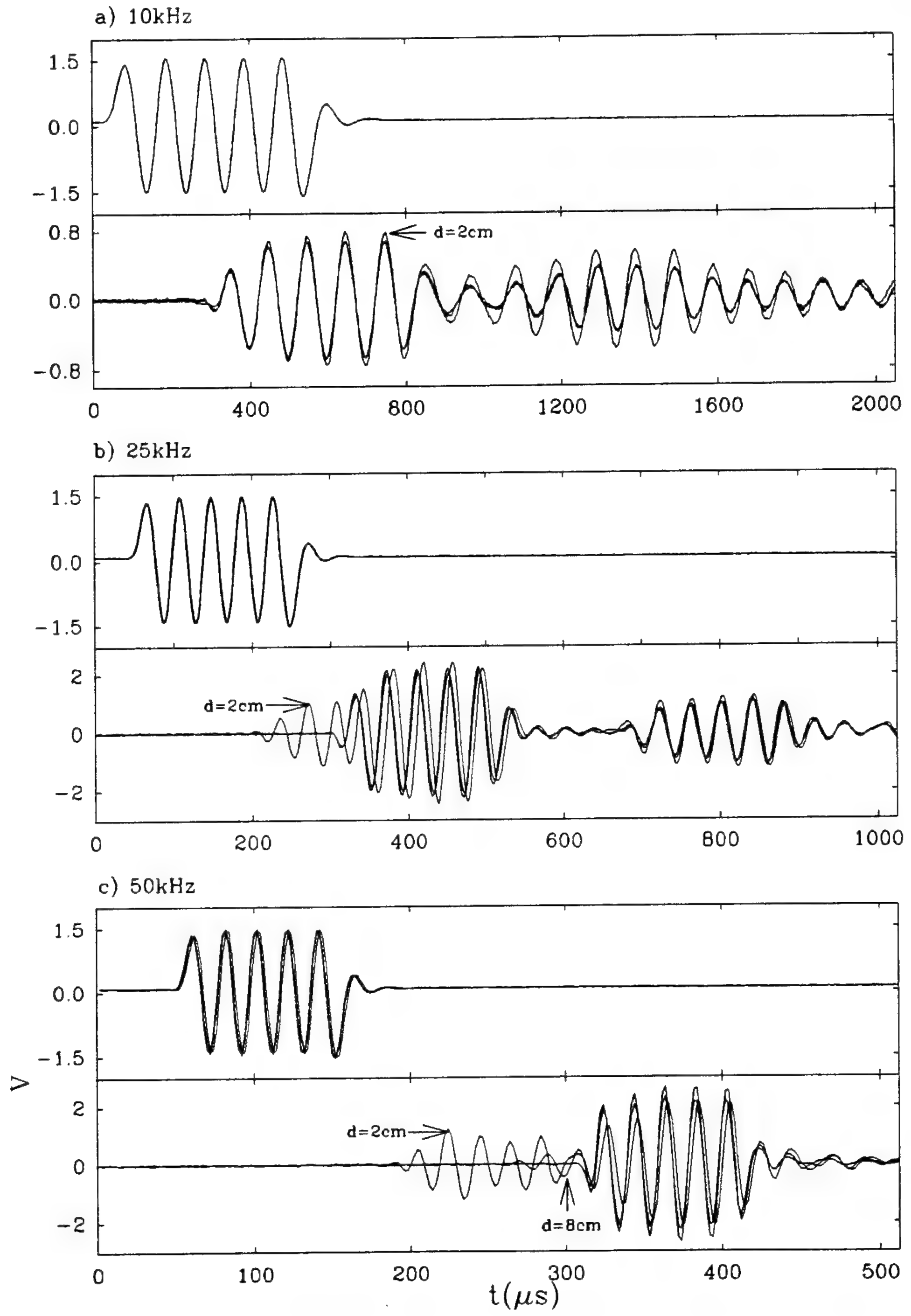


Figure 6.11: Pulse distortion from interference by a solid rod placed parallel and close to the direct propagation path. For each frame, the top plot is the transmit pulse and the bottom plot is the received pulse. a) 10kHz, b) 25kHz and c) 50kHz. At each frequency, three configurations were tested. 1) No rod, 2) d=2cm and 3) d=8cm. The separation d between the rod and the head of the hydrophones is indicated on the plots when the interference is noticeable.

6.9.2 Testing of interference caused by the acoustic support module

In the experiments described in the previous section, it was found that a separation of 8cm or more between the hydrophones and the main support rod would be sufficient to minimize the interference caused by the presence of the rod for frequencies below 25kHz. In the final design (figure 6.2), we used a separation of 10cm.

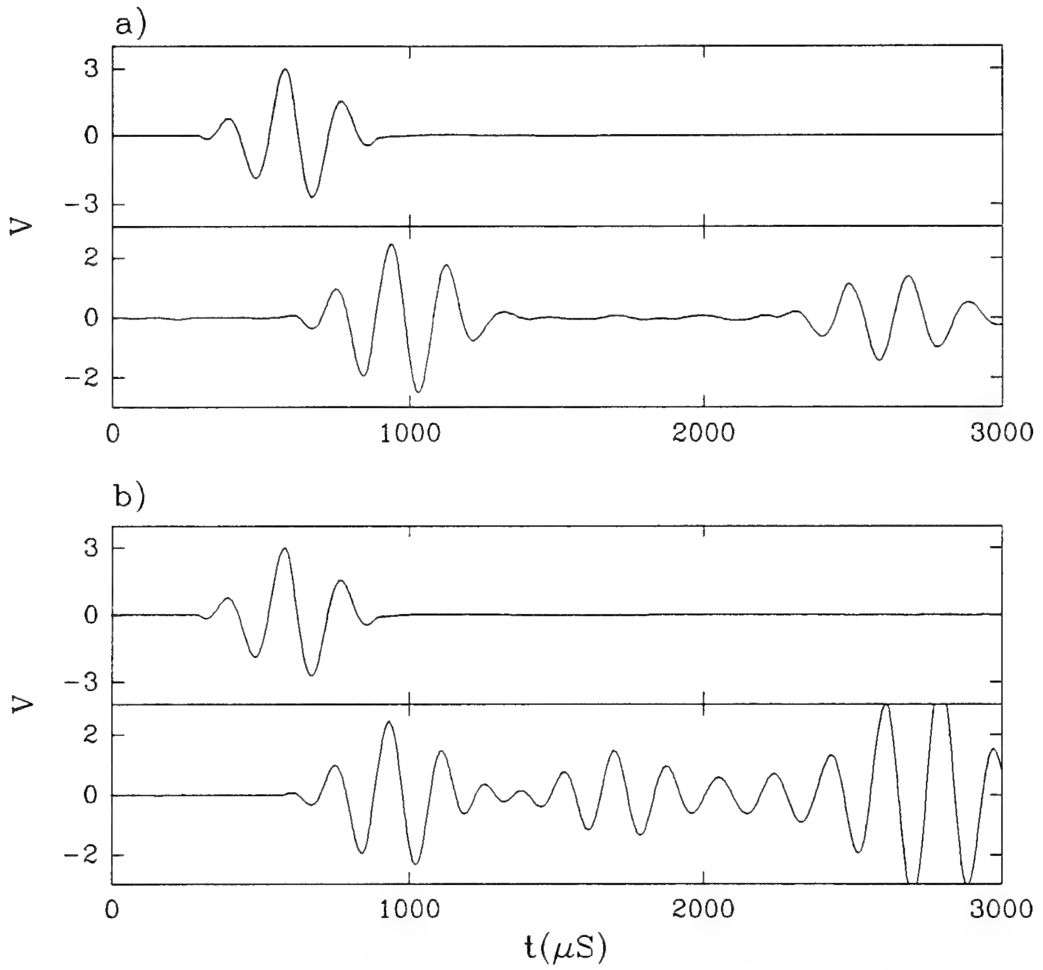


Figure 6.12: Test of acoustic interference by the support module at 5kHz. The experiments were conducted in a tank 3m in diameter and approximately 10m deep. a) The hydrophones were supported as in figure 6.10. b) The hydrophones were mounted on the support module of figure 6.2. The transmit and receive pulses are shown above and below respectively.

After construction, the hydrophone support module was tested to ensure that the final design did meet the criterion for minimal acoustic interference. The tests were conducted at the Scripps Institution of Oceanography in a deep water tank 3m in diameter and approximately 10m deep. The transmit and receive hydrophones were mounted on the support module which was suspended in the tank at a depth of 2 to 3m. The transmit-receive pulses were compared with the case when the hydrophones are only supported by long vertical rods as shown in figure 6.10. Since the hydrophones must be successively mounted and unmounted for these tests, the separation distance between the transmit and receive hydrophones will inevitably vary slightly between the two configurations tested. For this reason, we cannot expect to match the phase of the transmit and receive pulses. Instead, the focus is on comparing the shape and phase of only the receive pulses for the two configuration tested.

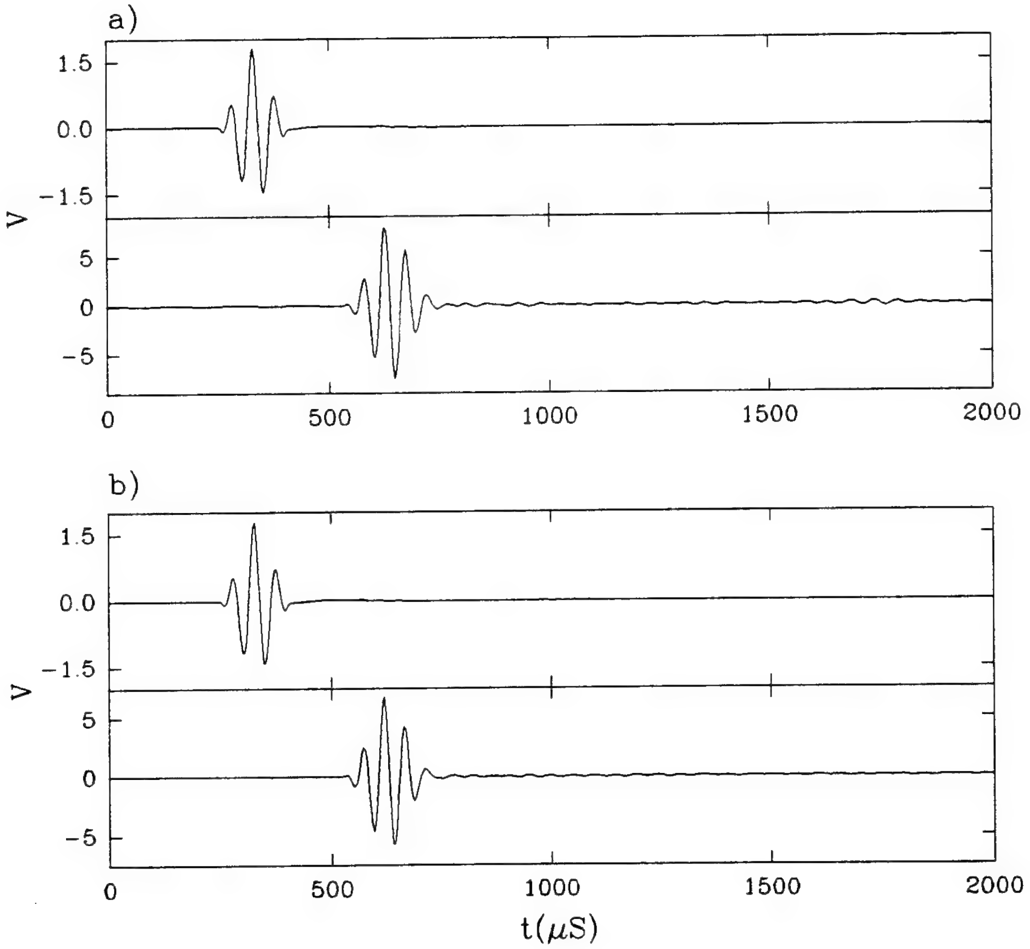


Figure 6.13: Test of acoustic interference by the support module at 20kHz. The experiments were conducted in a tank 3m in diameter and approximately 10m deep. a) The hydrophones were supported as in figure 6.10. b) The hydrophones were mounted on the support module of figure 6.2. The transmit and receive pulses are shown above and below respectively.

Figure 6.12 shows the results of a transmit-receive experiment at 5kHz without the support module (a) and with the support module (b). The pulse from the direct path shows no noticeable pulse distortions. Figure 6.13 offers similar conclusions for 20kHz pulses. The tests were also conducted at 10kHz and 40kHz and no pulse distortions, even at 40kHz, were observed.

6.9.3 Contamination of signal by bubbles adhering to the hydrophones

During the initial stage of the laboratory testing, the rusted metallic bottom of the water tank continuously generated very small air bubbles which adhered to both the transmit and receive hydrophones. This gave us an opportunity to investigate the performance of the transducers with bubbles adhering to their surface; a problem relevant to our acoustic measurements in a bubbly environment. A similar situation occurs when bubbles are trapped during the molding of a transducer. In such case, it is well known that the performance of the hydrophone will be affected at frequencies close to or at the natural resonance frequencies of the bubbles.

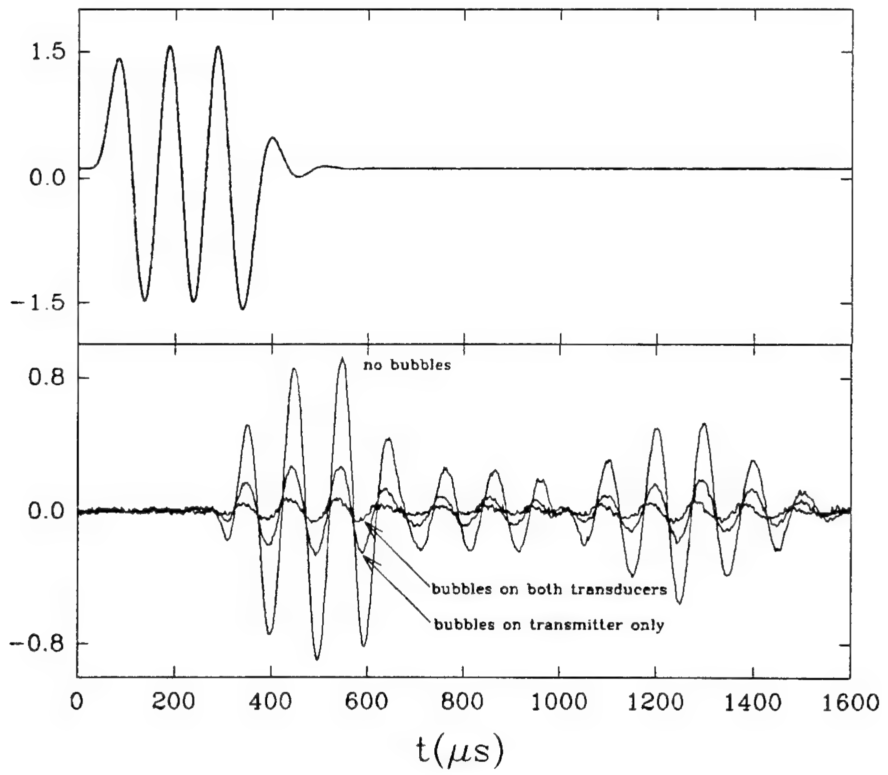


Figure 6.14: Pulse distortion caused by bubbles on the transducers. Transmit pulses (above) and receive pulses (below). The acoustic frequency is 10kHz.

A simple transmit-receive test of a 10kHz pulse was performed with bubbles on and off the transducers. The results are shown in figure 6.14. Clearly, phase shifts and significant signal attenuation occurs when bubbles are on either transducer. It should be pointed out that no attempt was made to actually measure the size of the bubbles on the transducers but a visual inspection suggested that the bubbles were in the range of 100μm to 1000μm in diameter. Recall that a bubble resonant at 10kHz would have a size of 660 μm diameter. Therefore, it may be that the signal distortion was caused by excited bubbles.

The more important question was whether such bubble adherence would actually occur at sea. Subsequent sea trials were to confirm that such effects did not occur. The long term (few hours) mean signal amplitude at low frequencies (5 and 10kHz) remained more or less constant within ±2dB confirming that bubbles did not build up on the hydrophones.

6.10 Cramer-Rao lower bound on the standard deviation of the time-delay estimate

The Cramer-Rao lower bound is a well known relationship in active and passive target localization systems [for a review see Quazi, 1988]. The relationship gives a lower bound on the standard deviation of the time-delay estimate given that the observation time T, signal to noise ratio SNR and signal bandwidth f_2-f_1 are known

$$\sigma_D \geq \left(\frac{3}{8\pi^2 T}\right)^{1/2} \frac{1}{\sqrt{SNR}} \frac{1}{\sqrt{f_2^3 - f_1^3}} \ . \qquad (6.8)$$

In essence, the Cramer-Rao lower bound states that for known signal characterisitic the minimum standard deviation on the time-delay estimates will be given by equation 6.8. Figure 6.15 shows how σ_D varies as a function of SNR for an observation time T and a bandwidth f_2-f_1 consistent with the 5kHz pulses used in the present study for measuring the speed of sound.

Figure 6.15 shows that measurements of the time-delay conducted in the laboratory will have standard deviation of ~3×10^{-7}sec and field estimates of the time-delay will have standard deviation of ~10^{-6}sec. Of course, this is for a single estimate of the time-delay. Statistical averaging would further decrease these standard deviations. The next section on the laboratory tests and §7 on the field measurements of the sound-speed will show that these lower bound estimates of σ_D were close to what was actually achieved with the acoustic system.

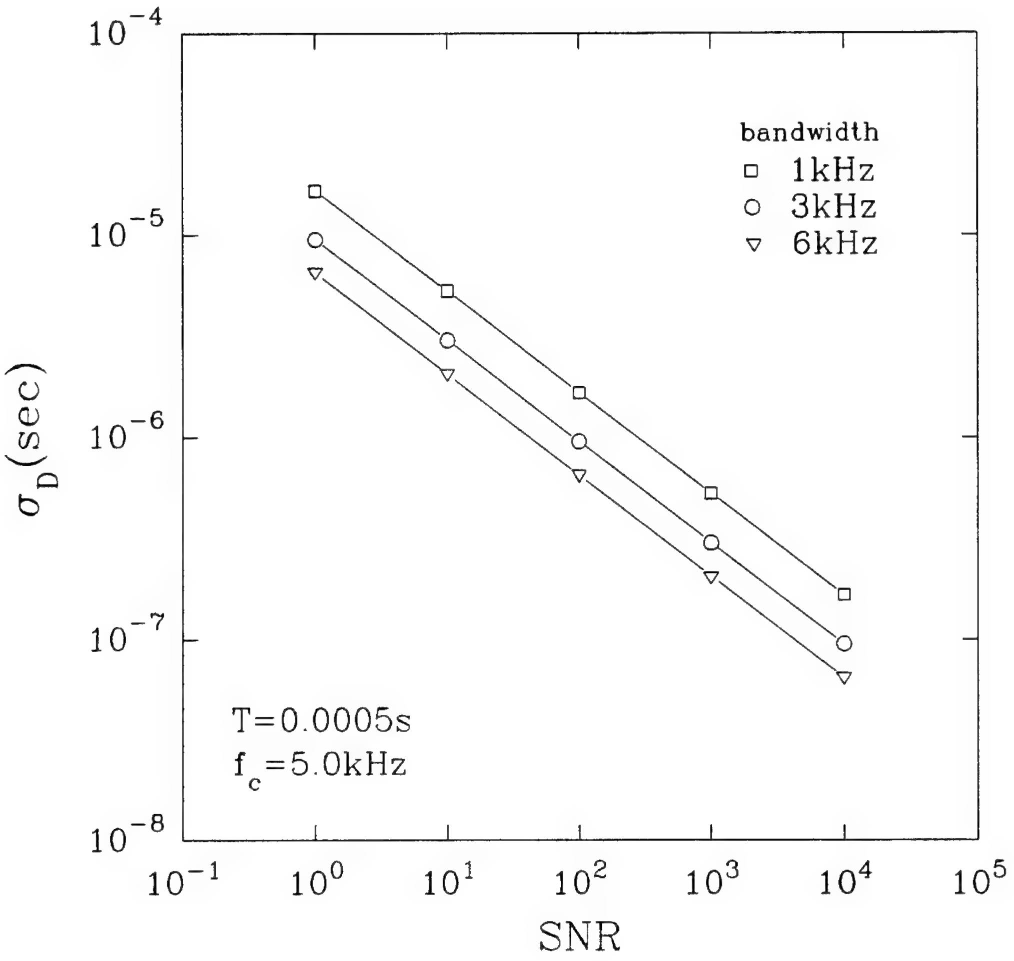


Figure 6.15: Cramer-Rao lower bound on the standard deviation of the time-delay estimate σ_D for an observation time T=0.0005s, a signal frequency $f_c=(f_1+f_2)/2$=5kHz and a signal bandwidth f_2-f_1=3kHz. Note that results are also shown for 1kHz and 6kHz bandwidths. For the laboratory, SNR~1000 and in the field SNR~100.

6.11 Laboratory tests

Two tests have been performed in the laboratory to investigate the performance of the sound-speed measuring system. The first one involved changing the separation distance between the transmit and receive hydrophone and measuring this change by monitoring the increase or decrease in the time-delay. The second test involved measuring the changes in sound-speed caused by changes in water temperature given that the relationship between temperature and sound-speed in water is well known.

6.11.1 Changes in time-delay caused by changes in the distance between the transmitter and the receiver

A controlled test of the acoustic instrument was performed by varying the distance between the transmit and receive hydrophone. For this test, the acoustic module was modified by cutting it in half along the center line (figure 6.16). The two halves were slid in a locking sleeve which permitted increasing or decreasing the separation distance between the hydrophones. The separation distance was varied between 400mm and 425mm. Small screws were installed on each half to act as reference marks for an exact measurement of the change in separation. The difference in separation Δs_a is with respect to the initial separation of 400mm. It was measured with a micrometer accurate to 0.03mm. With the acoustic system, the measured change in separation Δs_{td} was obtained by multiplying the measured change in time-delay Δt_{td} by the sound-speed $C(T^\circ)$ where the subscript "td" refers to time-delay and $C(T^\circ)$ is the speed of sound computed from temperature measurements (see next section for more details on the relationship between C and T°).

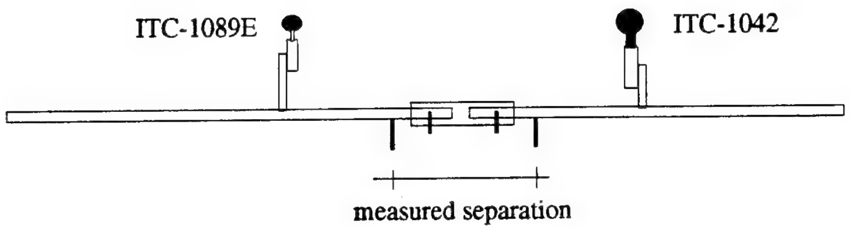


Figure 6.16: Modified support for varying the distance between the transmit and receive hydrophone.

Results for 20kHz and 5kHz pulses are presented in figure 6.17. Notice that the solid symbols have their vertical axis on the right hand side and that they represent the residual Δs_a-Δs_{td}. These results give solid support to the acoustic technique and they show that in ideal laboratory conditions, the typical errors over a 40cm path are within ± 0.5mm or ±0.13%. Accordingly, this suggest that sound-speed measurements made over a fixed distance of 400mm will have typical resolution of ±0.13%. This error is likely to increase at sea because of higher ambient noise levels, fluid velocity and buoy motion.

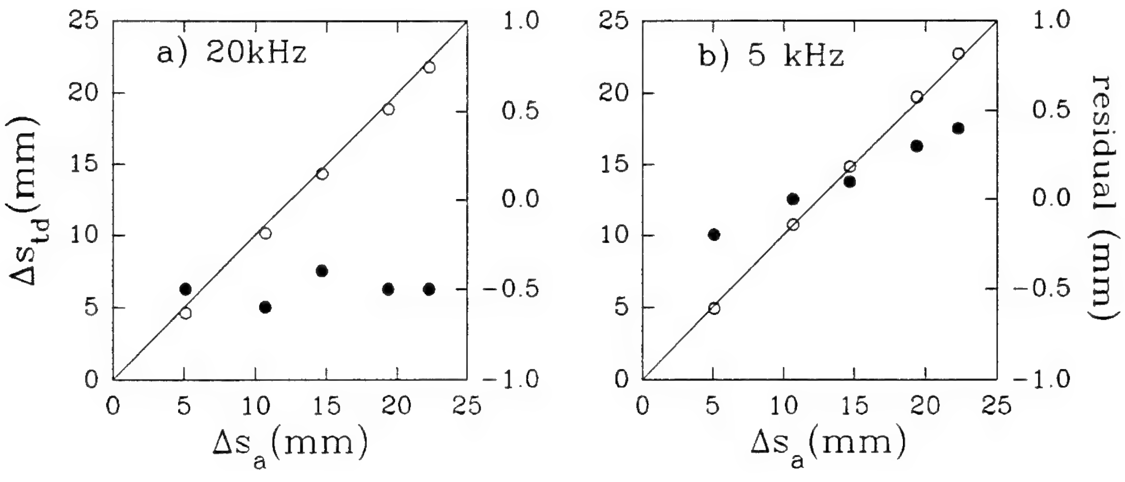


Figure 6.17: Changes in transmitter-receiver separation. The separation distance was varied between 400mm and 425mm. Only the change in separation Δs with repect to the 400mm reference is plotted. a) 20kHz, b) 5kHz. Δs_a measured with a micrometer. Δs_{td} measured from changes in time-delays. Actual measurements (o) and residual (•). Note that the vertical scale for the residual is on the right hand side of the graphs.

6.11.2 Changes in sound-speed due to changes in water temperature

It is well know that the speed of sound in water varies as a function of the salinity S, temperature T and depth of water d. For typical upper-ocean conditions, the dominant effect is temperature. For example, a one degree change in temperature will give rise to a ~3m/s change in the speed of sound. Of course, these changes are all small compared to sound-speed fluctuations caused by bubbles (see appendix A and B).

Many empirical equations have been developed to predict the speed of sound given a knowledge of S, T and d. Del Grosso [1974] has developed an elaborate equation applicable to wide ranges of salinity temperature and depths. Medwin [1975b] has simplified that equation for restricted ranges which are adequate for oceanographic applications

$$C(m/s) = 1449.2 + 4.6T - 0.55T^2 + (1.34-0.010T)(S-35) + 0.016d \qquad (6.9)$$

where 0≤T≤35°C, 0≤S≤45ppt and 0≤d≤1000m. This equation gives deviations from DelGrosso's equation [1974] on the order of tenths of 1m/s [Medwin, 1975b].

A test was performed in the water tank shown in figure 6.10. The temperature in the tank was varied by introducing warm or cold tap water. The temperature was measured with a Sea-Bird SBE 3 probe. After bringing the water tank to a new temperature, the tank was allowed to rest for 3 to 4 hours to allow for the fresh tap water to degas and for the temperature in the tank to become uniform. Each measurement consisted of an average of five sound-speed measurements separated by one second.

The results for two frequencies are shown in figure 6.18. The horizontal axis is the sound-speed inferred from equation 6.9 using temperature measurements and assuming S=0 and d=0.5m. The vertical axis is the actual measurement of sound-speed. Overall the results are very good. Clearly, the sound-speed measuring system has the ability to detect small changes in the speed of sound even at frequencies as low as 5kHz. The errors observed may have been caused by poor water quality or incomplete degassing of the water.

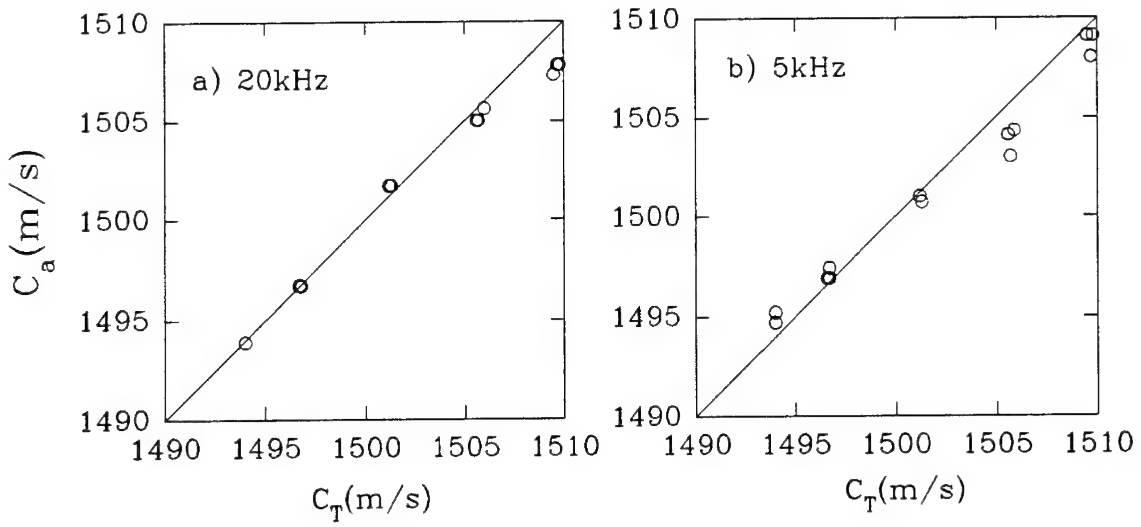


Figure 6.18: Changes in sound-speed caused by changes in temperature. C_T from temperature measurements and C_a from sound-speed measurements. a) 20kHz, b) 5kHz.

6.12 Lake test

A trial experiment was conducted in the Mystic lake in early September 1992. At that time of the year, New-England lakes still have their summer temperature stratification. Typical temperature differences between the surface and 10m below can range from 10°C to 20°C. This translates into 30m/s to 50m/s of sound-speed differences. This was therefore an ideal setting for testing the entire system with a large and stable sound-speed gradient. The buoy was positioned in water 20m deep and approximately 100m away from shore. The buoy was stripped of its flotation and gradually lowered in the water column by steps of ~0.5m. Five acoustic modules measured the sound-speed at different depths and a temperature probe was used along with equation 6.9 to infer the temperature-dependent speed of sound.

Results of the experiment are shown in figure 6.19 for two different acoustic frequencies. The solid line is a spline fit through the sound-speed data inferred from temperature measurement. The results show very good tracking of the sound-speed profile by the acoustic technique. The 20kHz data show less variations because the SNR is higher at that frequency and also, intuitively, one would expect the peak of the cross-correlation to be better defined as the acoustic frequency increases.

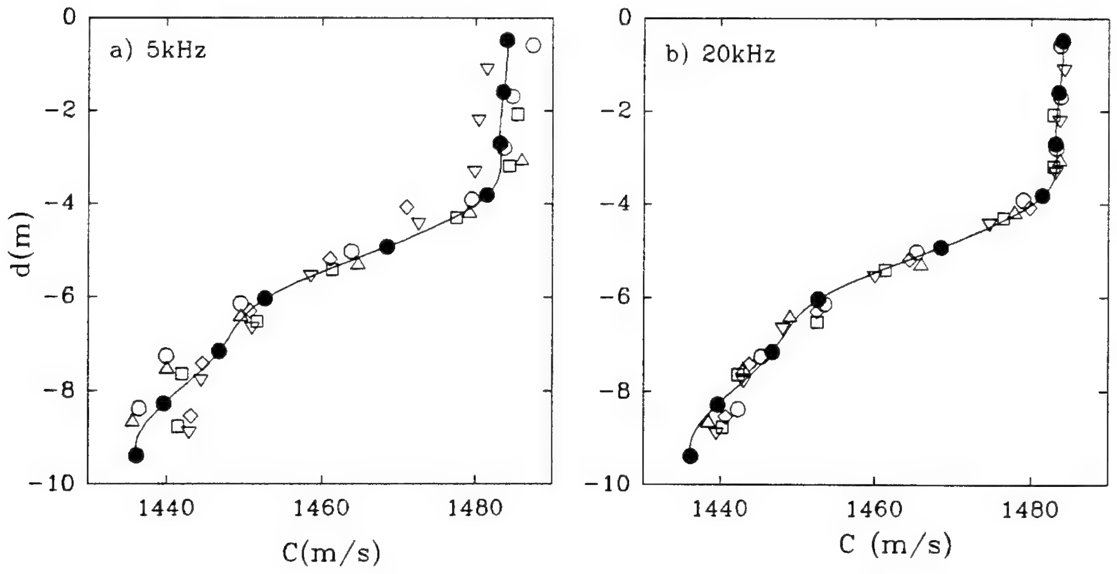


Figure 6.19: a) 5kHz, b) 20kHz. Changes in sound-speed caused by variations in the temperature with depth in the Mystic lake. Solid symbols (•) and spline fit are inferred from temperature measurements. Hollow symbols are measurements from different acoustic modules.

6.13 Discussion and summary

In this chapter, we have reported on the development of an effective technique for measuring the low-frequency (5kHz and above) speed of sound in the upper-ocean at depth of 0.5m and greater. Short acoustic pulses traveling over a 40cm path from a transmit to a receive hydrophone are used to make the measurements. Phase distortions caused by acoustic reflections from the surface or from nearby structural elements are separated from the direct path signal by carefully balancing the pulse length, acoustic wavelength, depth and hydrophones' separation.

The acoustic scattering characteristics of the support module for the transmit and receive hydrophones was studied extensively. All tests showed that the support generated negligible interference for frequencies at and below 40kHz.

A real time data acquisition and data processing system was implemented to process the acoustic pulses. Prior to digitizing the data, standard analog techniques were used to conditioned the signal. The data was sampled at 0.5MHz/ch and cross-correlated in real time by a DSP board to extract the time-delay and hence the sound-speed. The maximum throughput of the system was 24 sound-speed measurements per second including all overheads such as pulse transmission, acquisition and data storage.

In order to increase the resolution of the measurements, the cross-correlation function was interpolated in the neighborhood of its peak by a spline fit. This of course is only possible with signals possessing high SNR (in the present study, SNR~1000 in the laboratory and SNR~100 in the field). Laboratory data suggest that the interpolation scheme has given an order of magnitude increase in the resolution of the measurements.

Extensive laboratory testing of the sound-speed measuring system against controlled water temperature changes have shown that the technique can track the changes in sound-speed to within 2 or 3m/s. Similar experiments conducted in the Mystic lake during late summer, when the temperature profile of the lake is well stratified, have shown comparable resolution for a 60m/s spread in sound-speeds. All measurements were performed at 5kHz and 20kHz which showed that even at these low frequencies, accurate sound-speed measurements can be realized as long as the SNR is high and acoustic reflections are separated from the direct path signal.

Another test was also performed in the laboratory by accurately varying the separation distance between the transmit and receive hydrophones. The changes in separation were measured acoustically and compared with the measurements obtain with a micrometer. The results showed that the acoustic measurements had a maximum error of ±0.13% over a change in separation of 25mm and a total path length of 400mm. For a fix transmit-receive system, this translates into maximum errors of ±2m/s.

The laboratory and field tests have clearly demonstrated the capabilities of the low frequency sound-speed measurement technique. Before moving on to actual ocean measurements, it would be appropriate to discuss further validations of the technique with controlled sound-speed changes caused by a population of bubbles. At acoustic frequencies below bubble resonance, this is equivalent to validating Wood's equation (appendix A) which expresses the speed of sound as a function of the void-fraction. These kind of experiments have been performed at intermediate (10^{-4} to 1) [Silberman, 1957] and high (10^{-2} to 1) [Karplus, 1958; Ruggles, 1987] void-fractions and they have confirmed the validity Wood's equation. However, no experiments have been performed at the low void-fraction range (10^{-8} to 10^{-5}) mainly because of the great difficulty in measuring the void-fraction in that range and in generating a steady and controlled bubble population at such small air concentration. On their own, these difficult measurements require further research and this is why they were not considered for testing the present sound-speed measurement system.

Chapter 7

Sound-speed and attenuation measurements near the ocean surface

In this chapter, we report on the results of two field experiments conducted in the fall of 1992 on the measurement of sound-speed and attenuation near the ocean surface. The buoy and the instrumentation developed for these measurements were described in §6. In the first part of this chapter, we analyze several time-series of sound-speed measurements taken at various depths. In particular, we investigate the dependence of the signal with depth, wind speed and significant wave height (SWH). The spectral characteristics of the signals, their coherences and their probability distributions are also studied.

In the second part of the chapter, we look at the dependence of the sound-speed and attenuation measurements with frequency. These measurements lead us to investigate the possibility of extracting bubble population information from measurements of the complex sound velocity.

7.1 The site of the experiments

The first experiment was conducted during the first week of October 1992 on the R/V Sproul[1] off the coast of California near San Diego (SD). The winds were relatively weak during the period of this experiment and therefore it was decided to position the ship in the Santa Cruz Channel, between the islands of Santa Cruz and Santa Rosa, in order to benefit from the wind-tunneling effect caused by the geometry of these islands. The maximum steady wind speed recorded by the ship's anemometer for the entire period of the experiment was approximately 8m/s. Thus, the wind conditions for the San Diego experiment varied from very calm to moderate and the waves were fetched-limited by the length of the Santa Cruz Channel which measures approximately 12km. The water depth in the channel is approximately 60m. For this experiment, we report on the results from approximately seven hours of data acquired in the Santa Cruz Channel and three hours

[1] The R/V Sproul is a ship of the Scripps Oceanographic Institution. It measures 38m long and it has a 2.9m draft.

acquired at other positions along the coast in very calm sea-states (wind speeds less than 3m/s).

The second experiment was conducted during the first two weeks of November 1992 on the R/V Asterias[2] in Buzzards Bay off the coast of Woods Hole (WH), Massachusetts. The purpose of the experiment was to complement the data set acquired during the San Diego experiment with data at higher wind speeds ($U \geq 8$m/s). Accordingly, the ship only made short day trips into Buzzards Bay when the wind conditions were satisfactorily high. However, the wind conditions for the two weeks of the experiment were unfortunately not higher than 8m/s (steady) and therefore the data set for the Woods Hole experiment is limited to five hours of data at winds varying between 6 to 8m/s and significant wave height between 0.2m to 0.65m. All of the measurements reported from this experiment were acquired close to the Buzzards Bay entrance tower where the water depth is approximately 20m. A MiniSpec wave height sensor (Coastal Leasing, Cambridge, MA) was suspended between two legs of the tower in order to measure the hydrostatic pressure from which the wave height was calculated.

7.2 The experimental procedure

On a typical deployment, the buoy and its instrumentation were kept approximately 100m to 130m from the ship and the boat was maneuvered such that the buoy was kept on the port or starboard side of the ship as much as possible. Thus, the ship's wake was kept away from the buoy. The tether was constantly monitored and kept slack by maneuvering the boat accordingly. Hence, the buoy was essentially drifting under its own drag and the drag of the cable and floats that linked it to the ship.

The deployments generally lasted between 1 to 3 hours and during that time, the sound-speed and the attenuation profiles were measured at different frequencies for a period of 10 to 20min at each frequency. The selected frequencies used for the experiments are given in table 6.2. The sound-speed was sampled at a rate of 2Hz/ch and a small fraction of the data set was sampled at the maximum rate of 4Hz/ch. Note that the time-series were composed of continuous 80s data segments separated by 10s gaps during which the raw and the processed data were written to optical disk. In the

[2] The R/V Asterias is a ship of the Woods Hole Oceanographic Institution. It measures 14m long and it has 1.4m draft.

presentation of the results, the small 10s gaps are ignored when plotting 20min time-series but they are taken into account and shown when plotting shorter time segments.

Short 5 to 10min ambient sound records were also acquired approximately once or twice per hour. During those periods, all 6 ITC-1089 hydrophones were sampled along with the ITC-8181A hydrophone located at the bottom of the buoy. The sample rate of the hydrophones was 41.7kHz/ch. The data was sampled in bursts of 24.6mS with a burst repetition rate of 5Hz. Finally, the acoustic signature of a broad-band acoustic signal transmitted by the uppermost ITC-1042 hydrophone was also recorded with exactly the same sampling arrangements as for the ambient sound.

7.3 Time-series and signal characteristics

The sound-speed anomaly is defined as

$$\Delta c = c_w - c \qquad (7.1)$$

where c_w and c are the speed of sound in bubble-free water and in the bubbly mixture respectively. The bubble-free sound-speed c_w is obtained from measurements at sea during very calm conditions when there is no breaking. Note that the temperature is continuously recorded during the experiment in order to compensate for the variations of c_w with temperature. The determination of the sound-speed is therefore equivalent to the determination of the sound-speed anomaly since the knowledge of c_w permits the computation of one from the measurement of the other. Hence, both expressions are used interchangeably in the context of this work.

We begin the survey of the data set by first showing a typical time-series of the sound-speed anomaly in 8m/s wind conditions. Figure 7.1 shows time-series of the sound-speed anomaly for 10kHz acoustic pulses taken simultaneously at depths of 0.5, 0.75 and 1.0m. The signals at different depths are clearly correlated which would suggest that fluctuations are caused by well defined bubble clouds which drift by the sensors. The magnitude of the anomalies is also found to rapidly decrease with depth which is consistent with the fact that the bubble density is expected to be the highest near the surface. Because of the moderate wind conditions, the occurrences of large excursions in the signal are relatively infrequent and usually followed by periods where the signal returns back to the noise level. The fluctuations themselves are typically very large and they clearly dominate the long-term average at all depths. Note that the corresponding void-fraction based on an approximation given in appendix A (equation A.7) is shown on the vertical axis on the right-hand side.

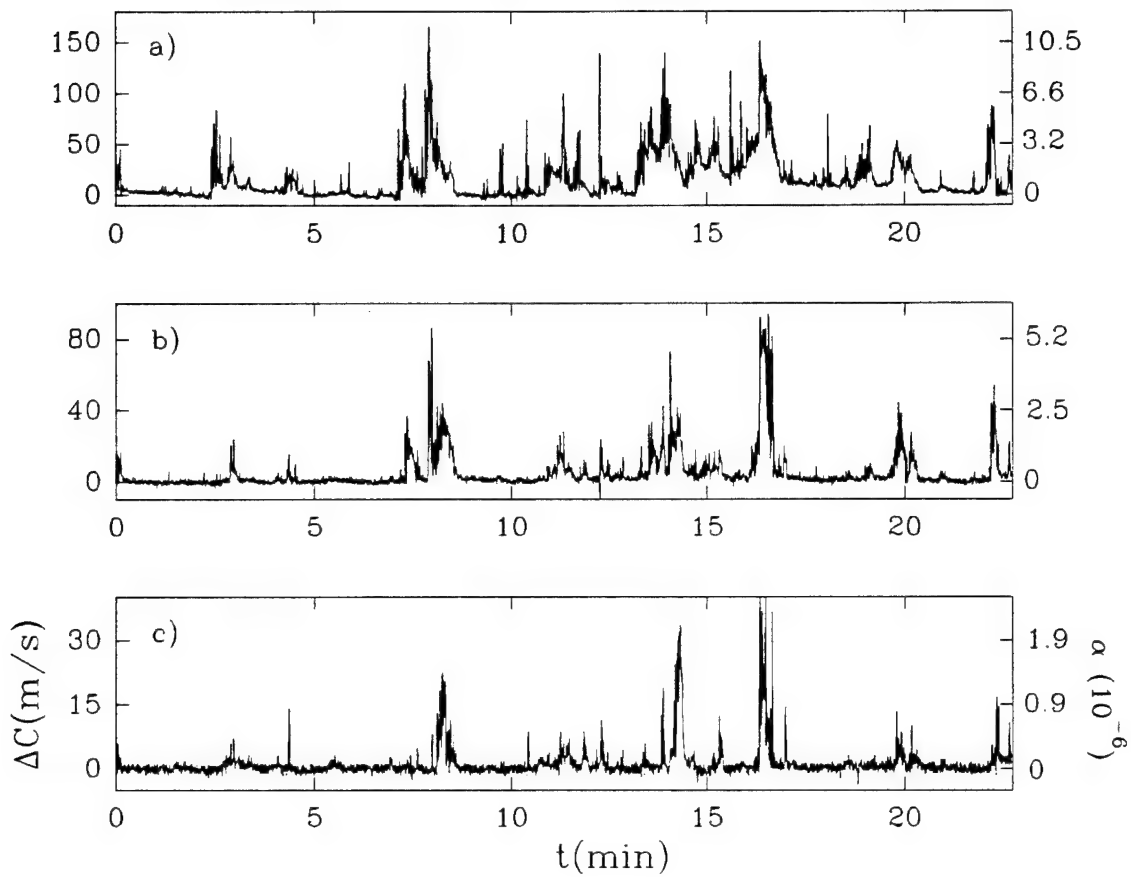


Figure 7.1: Sound-speed anomalies at depths a)0.5m, b)0.75m, and c)1.0m. Note the different vertical axes for all three plots. The axes on the right-hand side show the corresponding void-fraction. The frequency of the acoustic pulses was 10kHz. The wind speed and the SWH were 8m/s and 0.45m respectively. Sampling rate was 2Hz/ch. Woods Hole data.

Figure 7.2 shows time-series similar to the ones shown in figure 7.1 with the difference that the acoustic pulses were at 5kHz instead of 10kHz. The data show similar structures which are clearly correlated with depth and which also decrease very rapidly in amplitude with an increase in the depth of the sensors.

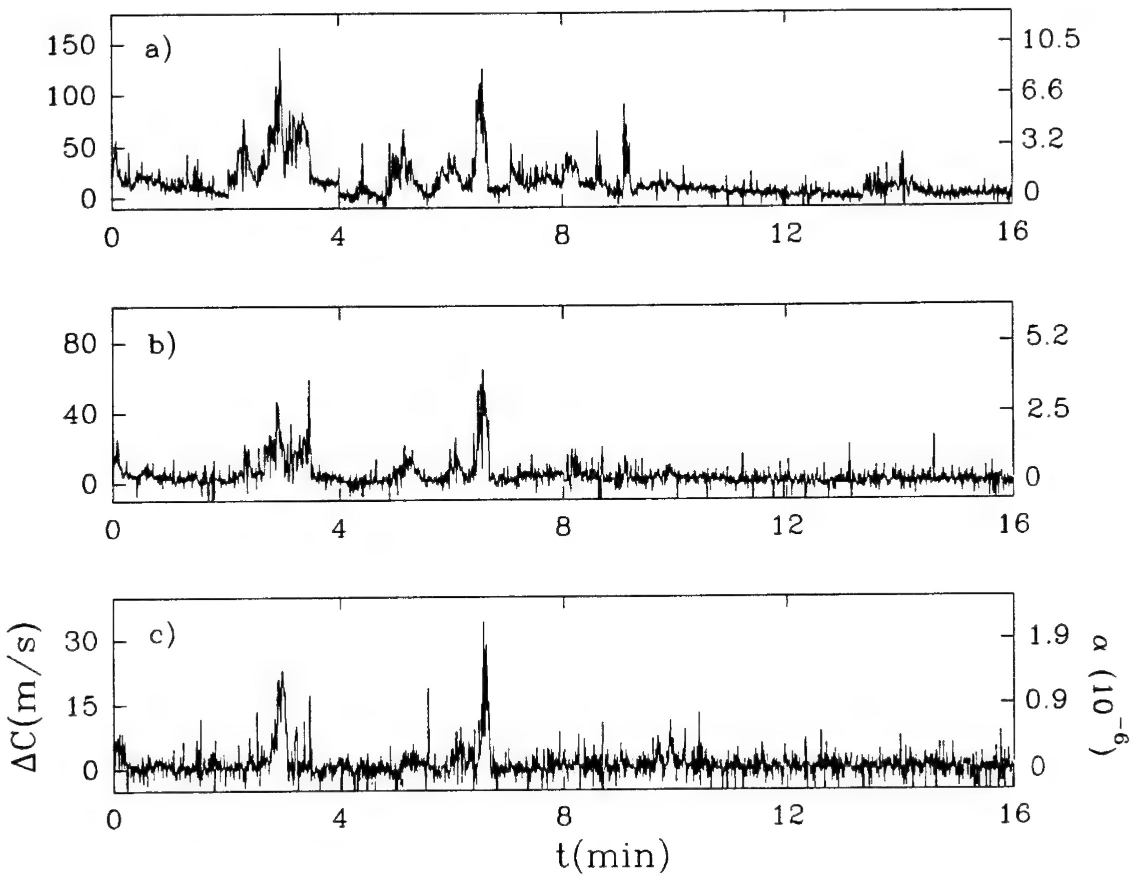


Figure 7.2: Sound-speed anomalies at depths a)0.5m, b)0.75m, and c)1.0m. Note the different vertical axes for all three plots. The axes on the right-hand side show the corresponding void-fraction. The frequency of the acoustic pulses was 5kHz. The wind speed and the SWH were 8m/s and 0.45m respectively. Sampling rate was 2Hz/ch. Woods Hole data.

The root-mean-square (rms) noise at 10kHz was calculated to be 0.7m/s and the peak-to-peak noise level of the measurements is approximately ±2m/s which is consistent with the laboratory tests of §6.10. At 5kHz, the rms noise is 2.7m/s and the peak-to-peak noise is approximately ±5m/s. The higher noise level at 5kHz is caused by ship-noise which is typically high at frequencies below 1kHz and decreases with an increase in frequency beyond 1kHz.

For both the Woods Hole and the San Diego experiment, we have consistently observed the presence of a signal above the noise threshold down to a depth of 1.0m for the highest wind conditions attained in both of these experiments (~8m/s). However, very few occurrences of signals above the noise level were observed at a depth of 1.5m in the entire data set and none at a depth of 2.0m and greater.

A short segment of data was extracted from the time-series in figure 7.1 in order to show the finer structure of the signal (see figure 7.3). When the sound-speed anomaly rises above the noise level in figure 7.3, the signals at all depths appear to contain fluctuations at surface wave frequencies (periods of 4 to 5sec) which ride on lower frequency signals. The spectral analysis of the next section will reveal more insight on the frequency content of the signal.

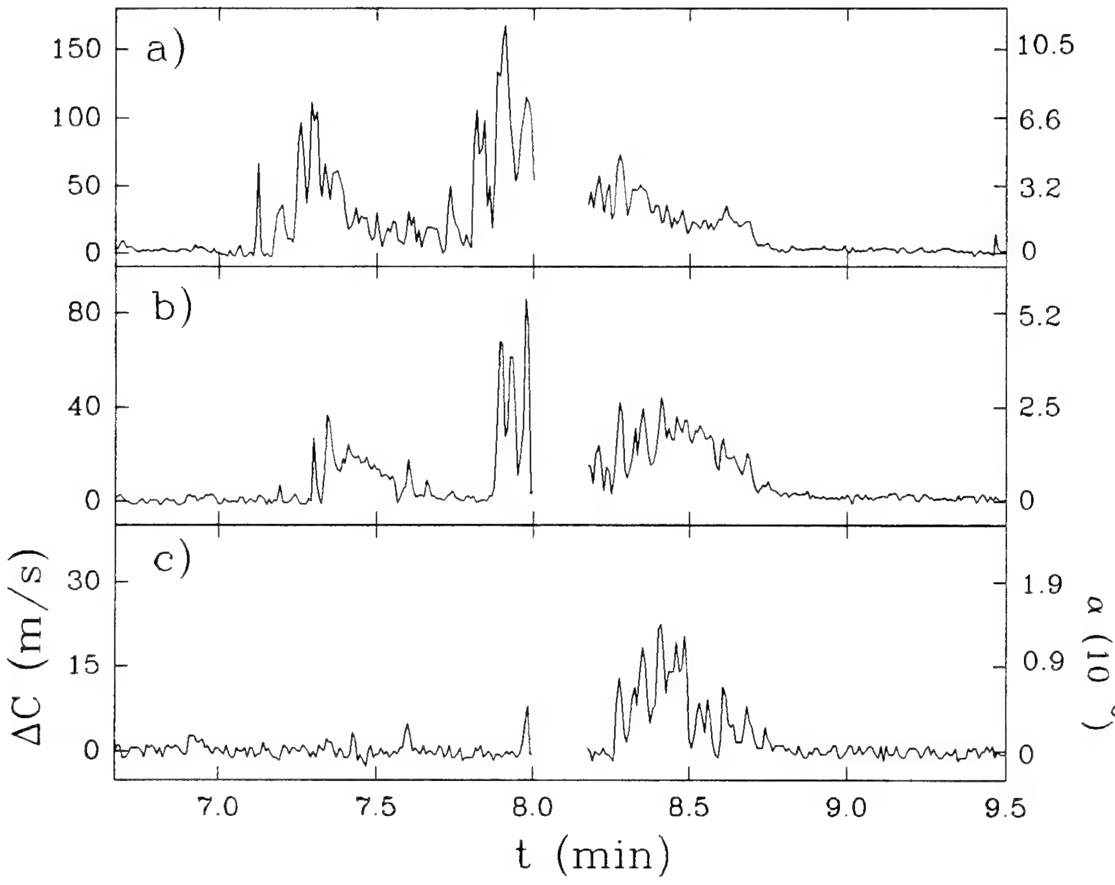


Figure 7.3: A short enlarged segment of the time-series shown in figure 7.1. The respective plots correspond to depths of a)0.5m, b)0.75m and c)1.0m. Notice the high-frequency signal with a period of 4 to 5s which rides on lower frequency signals. The short 10s gap in the middle corresponds to a period of data storage.

7.3.1 Spectral analysis

The frequency spectrum of the time-series in figure 7.1 was computed along with those of two other 20min time-series taken during the same deployment at the same acoustic frequency and at similar wind speed and significant wave height. The remainder of §7.3 will exclusively focus on the signal characteristics of these three 20min records. The three time-series were subdivided into a total of 42 80s segments which were windowed using a Hanning window. The 42 individual spectra were then ensemble averaged to yield the spectra of figure 7.4 for four different measurement depths. As expected, the energy in the spectra decreases with depth until it reaches the noise level for the lowermost spectrum which is for measurements taken at a depth of 2.0m.

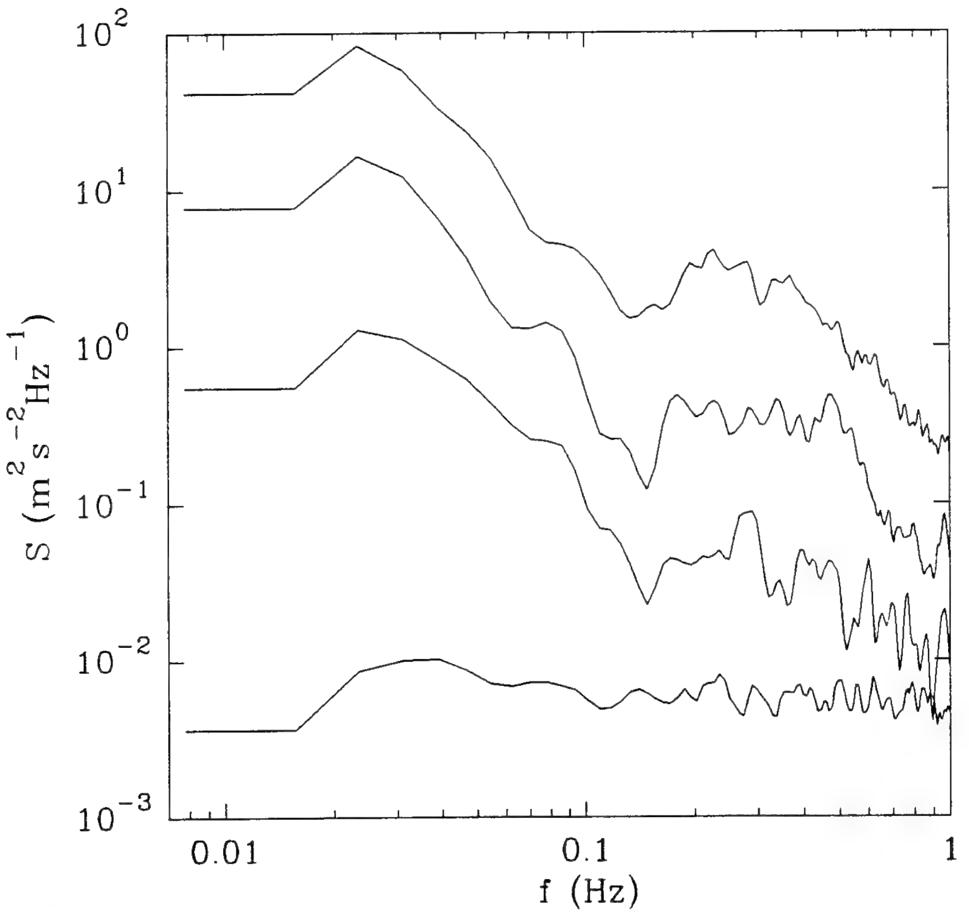


Figure 7.4: Frequency spectra of the sound-speed anomalies at four different depths. From top to bottom, the various solid lines are for measurements at 0.5m, 0.75m, 1.0m and 2.0m respectively. The spectrum a 2.0m is representative of the noise level. The frequency of the acoustic pulses was 10kHz. Wind speed and SWH were 8m/s and 0.45m respectively. Woods Hole data.

For the other three shallower depths, most of the energy in the signal is contained at frequencies much lower than the dominant frequency of the waves (see below). The energy in this low-frequency region is due to the large and relatively slow fluctuations present in the time-series and caused by the passage of bubble clouds. Note that the low-frequency peak in the spectra at ~0.02Hz is caused by the use of a short 80s record length to compute the individual spectrum. Thus, the spectral resolution for frequencies below 0.02Hz may not be reliable and the peak could be at a lower frequency.

At higher frequency, there is a distinct peak in the spectra of figure 7.4 at approximately 0.25Hz. Figure 7.5 compares the sound-speed anomaly spectrum at 0.5m (figure 7.4) with the wave height spectrum measured at the Buzzards Bay entrance tower which was located never more than 3km away from the ship during the deployments. Figure 7.5 shows that the peak at ~0.25Hz corresponds to the peak of the dominant waves. This can be explained by the fact that the buoy response always lags, to some extent, the changes in the wave field and this causes small variations in the vertical and horizontal positions of the sensors with respect to the orbital motion of the water located at the same depth as the sensors. Thus, bubble clouds are advected laterally and vertically past the sensors with a velocity due in part to the orbital motion of the waves. Even if the buoy was perfectly following the water surface, such signals would still be generated because the orbital motion of the wave decays with depth while the sensors maintain a fixed distance from the buoy at the surface.

Since there are important horizontal and vertical variations in the sound-speed anomaly (note that variations in time correspond, to some extend, to variations in the horizontal direction since bubble clouds drift by the sensors), the advection generated by the difference in orbital motion generates a signal at the dominant wave frequency. It was estimated from the underwater video measurements, that the buoy tracked the water surface to within ±10cm. It was also estimated[3] that the differences in orbital motion between the surface and 1.0m depth is on the order of 5 to 10cm. Thus we can estimate that the buoy undergoes lateral and vertical excursions, beyond those of the orbital motion of the wave, on the order of 5 to 10cm. The sound-speed anomaly signal generated by the wave field show that the sound-speed field is highly variable over small excursions on the order 5 to 10cm.

[3] Assume a wave period T=4sec and a significant wave height H=0.5m. The diameter of the orbital motion at the surface will be equal to H and at a depth of 1.0m it will be equal to H $e^{-kz} = 0.78H$ where $k=2\pi/\lambda$, $\lambda=1.56T$ and $z=-1.0$m.

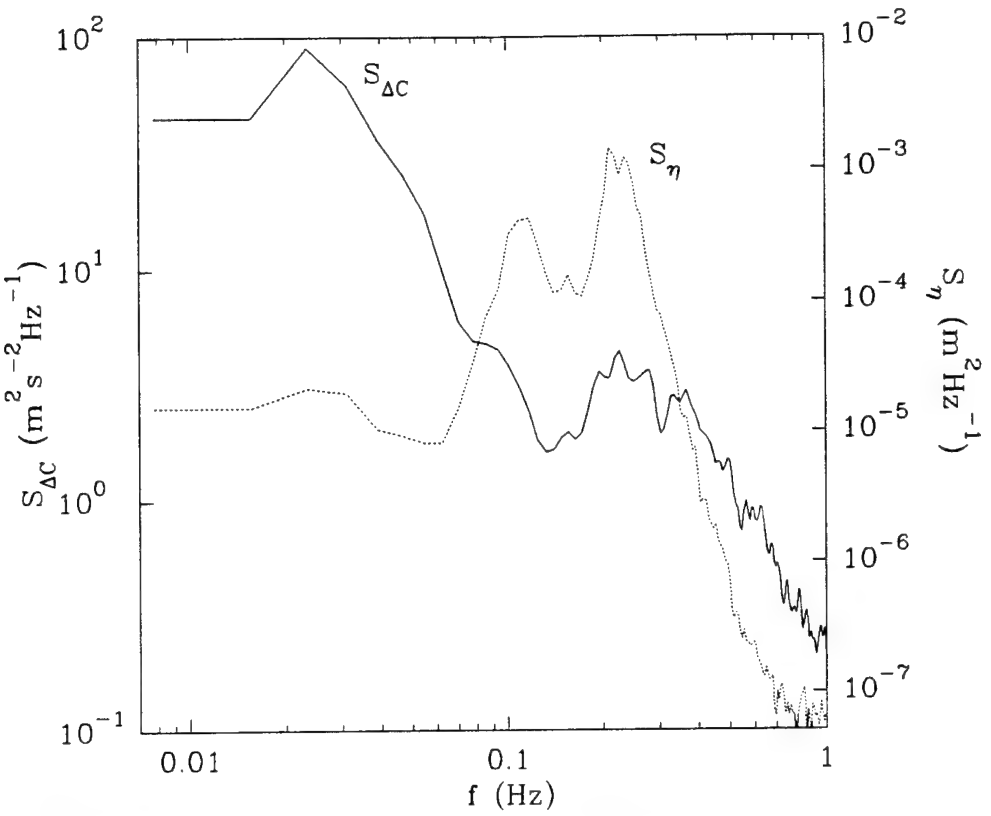


Figure 7.5: Solid line is the frequency spectrum of the sound-speed anomaly at 0.5m (from the upper curve in figure 7.4) and the dashed line is the frequency spectrum of the wave height measured at the Buzzards Bay entrance tower.

Figure 7.6 shows another series of averaged spectra similar to the ones shown in figure 7.4. The interesting difference is that the peak of the wave generated sound-speed fluctuations is at a higher frequency (~0.4Hz). The three 20min time-series used to compute the spectra of figure 7.6 were from the San Diego experiment where the fetch was limited. Indeed, the San Diego data were acquired in the 12km long Santa Cruz Channel where the winds were being funneled by the geometry of the surrounding islands. The shortest and longest fetch were estimated to be approximately 3 and 9km respectively. The dominant wave frequency was estimated by using a simple wave forecasting equation along with measurements of the wind speed and knowledge of the fetch [U.S. Army Corps of Engineers, 1984, pp.3-48]. The lower and upper bound dominant wave frequencies corresponding to a 3 and 9km fetch were estimated to be 0.4 and 0.55Hz respectively and these results are plotted as vertical lines in figure 7.6. As in figure 7.5, the peak of the dominant waves and the peak in the spectra of the sound-speed anomalies matched well. This shows again that a relatively strong signal is generated by the wave field which causes modest horizontal and vertical displacements of the sensors with respect to the ambient fluid on the order of 5 to 10cm.

It should be pointed out that the above results are not restricted to a 10kHz acoustic pulse frequency and that similar results were obtained with 5kHz pulses although the higher noise level at that frequency made it more difficult to detect the wave generated peak in the spectra of the sound-speed anomaly.

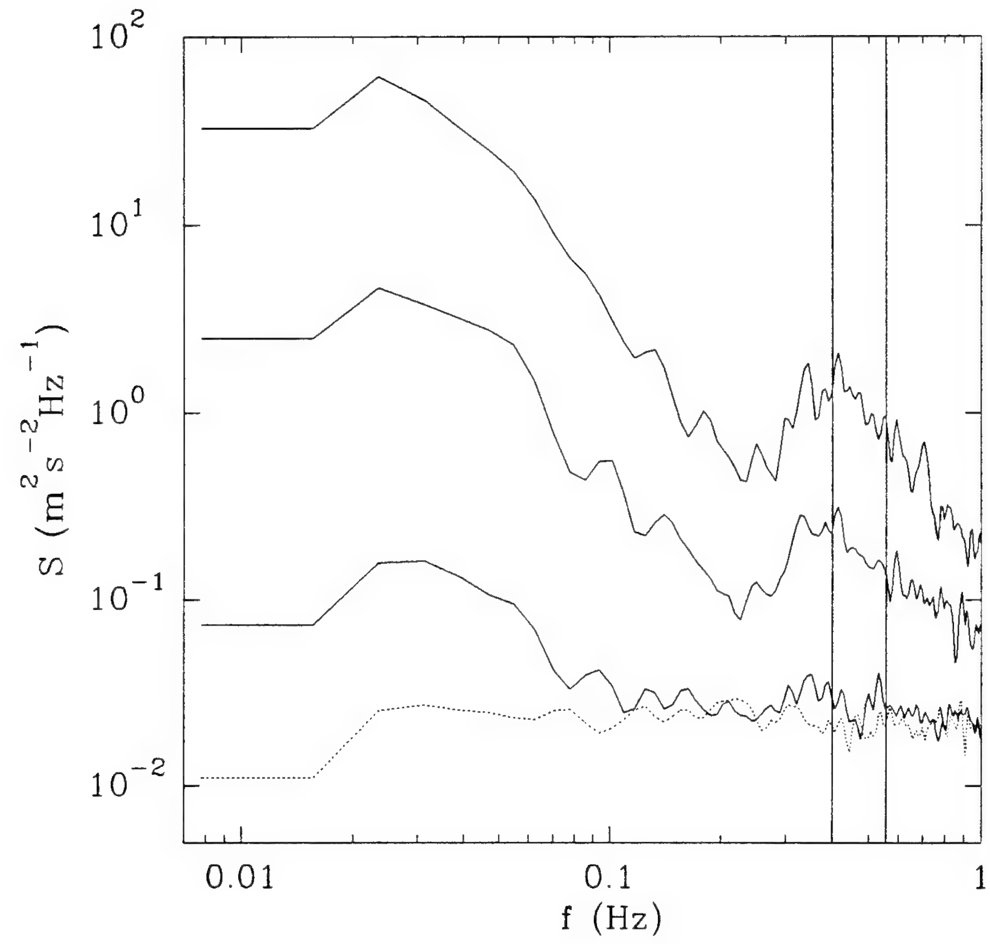


Figure 7.6: Frequency spectra of sound-speed anomalies at 4 different depths. From top to bottom, the various lines are for measurements at 0.5m, 0.75m, 1.0m and 2.0m (dashed line) respectively. The two vertical lines are the lower and upper-bound estimates of the dominant wave frequency. The spectrum at 2.0m (dashed line) is representative of the noise level. The frequency of the acoustic pulses was 10kHz. Wind speed was 5.5 to 6.5m/s. San Diego data.

7.3.2 Correlation and coherence

The correlation and coherence between measurements realized at different depths were computed for the same three 20min records used in the spectra of figure 7.4. To facilitate the identification of the various depths on the figures, we adopt the convention that 0.5m, 0.75m and 1.0m correspond to the depth numbers 1, 2 and 3 respectively. Thus, the correlation coefficient $\rho_{1\text{-}2}$ is obtained by cross-correlating the measurements at 0.5m and 0.75m. Figure 7.7a shows the correlation coefficient ρ as a function of the lag τ for lags close to zero. The data show that measurements at various depths are significantly correlated and that the correlation diminishes as the distance between the sensors increases. This is mostly due to the fact that the deeper sensors miss the smaller sound-speed anomalies. Note that the time-series were low-pass filtered at 0.1Hz before being cross-correlated. This was done to eliminate the wave generated signal which is in-phase at all depths and which generated a small peak in the correlation function at zero lag. With filtering at 0.1Hz, the contribution of the surface-wave generated signal is eliminated and the remaining correlation function becomes characterisitc of the low-frequency signal caused by bubble clouds drifting by the sensors.

The data also show that the highest correlation are at some small positive lag which increases with sensor separation. A positive lag corresponds to a signal arriving at the shallower sensors first. Two possible mechanisms can be suggested for the generation of this lag. The lag could be caused by the finite vertical penetration speed of the bubble plumes entrained by breaking waves. The lag could also be caused by the shape of the bubble clouds which drift pass the sensors. Indeed, many sonar studies of bubble clouds at sea [see Thorpe, 1992 for a review] have shown that the horizontal dimensions of the clouds decrease with depth. Consequently, a drifting cloud should reach the uppermost sensor of the buoy first and, as drifting progresses, the signal will eventually appear at the other sensors with a slight delay which will increase with depth. It is not possible at this time to separate the individual contributions of the two mechanisms discussed above.

The coherence shown in figure 7.7b shows that measurements at various depths are most coherent at very low-frequencies and that the coherence decreases as the separation between the sensors increases. Again, this is explained by the fact that the deeper sensors do not detect many of the smaller sound-speed anomalies. Also notice the relatively high coherence in some frequency bands near the surface wave frequencies (~0.25Hz).

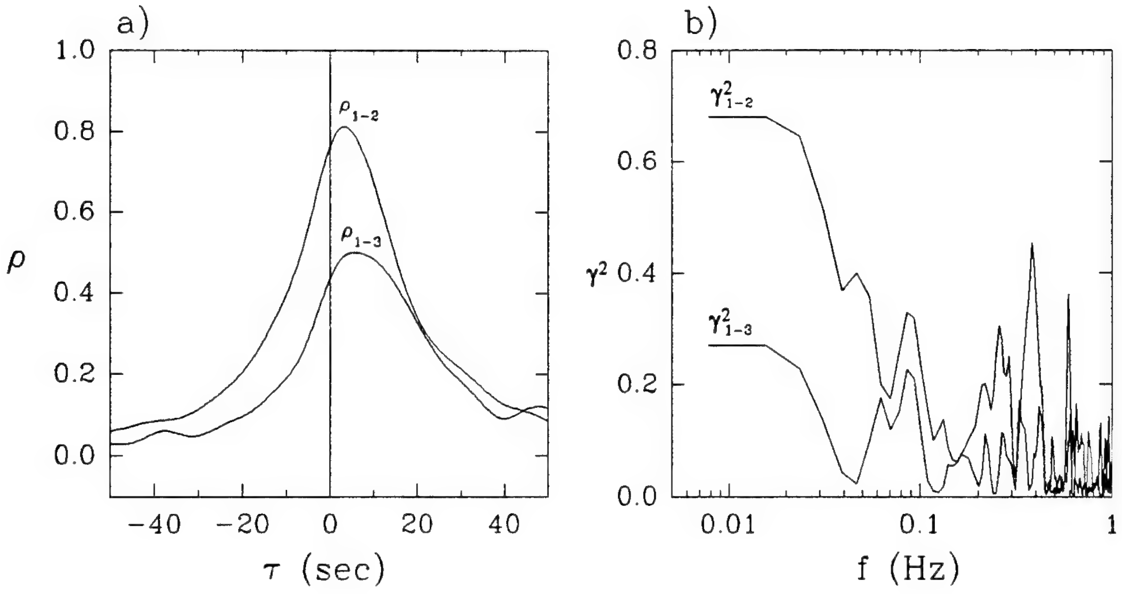


Figure 7.7: a) Correlation coefficient between measurements at 0.5m and 0.75m (ρ_{1-2}); and between measurements at 0.5m and 1.0m (ρ_{1-3}). b) Coherence between measurement at 0.5m and 0.75m (γ^2_{1-2}); and between measurements at 0.5m and 1.0m (γ^2_{1-3}). The average of three 20min time-series were used for both plots. The time-series were digitally low-pass filtered at 0.1Hz for the results shown in a) in order to eliminate the high-frequency wave-generated signal which is in-phase at all depths and which distorted the peak near zero lag. Wind speed and significant wave height were 8m/s and 0.45m respectively. Woods Hole data.

7.3.3 Probability distribution

The probability distribution of the sound-speed anomaly at a depth of 0.5m for the same three 20min records is shown in figure 7.8a. The shape of the distribution is close to exponential except close to zero sound-speed anomaly where the noise of the measurements dominates the distribution and forces it to more closely fit a normal distribution. The data in figure 7.8a are plotted on a log-linear scale in figure 7.8b along with data at three other depths. The distribution at a depth of 2.0m is representative of the measurement noise. Away from the noise region (say $\Delta C > 2$m/s), the distributions at all depths are exponential with the slope of the distributions steepening with depth. This indicates that the sound-speed anomalies above the noise of the instrument are becoming less frequent and of lesser magnitude as the depth increases. The exponential nature of the distributions also indicates that the frequency of occurrence of sound-speed anomalies decreases very rapidly with an increase in the magnitude of the anomaly.

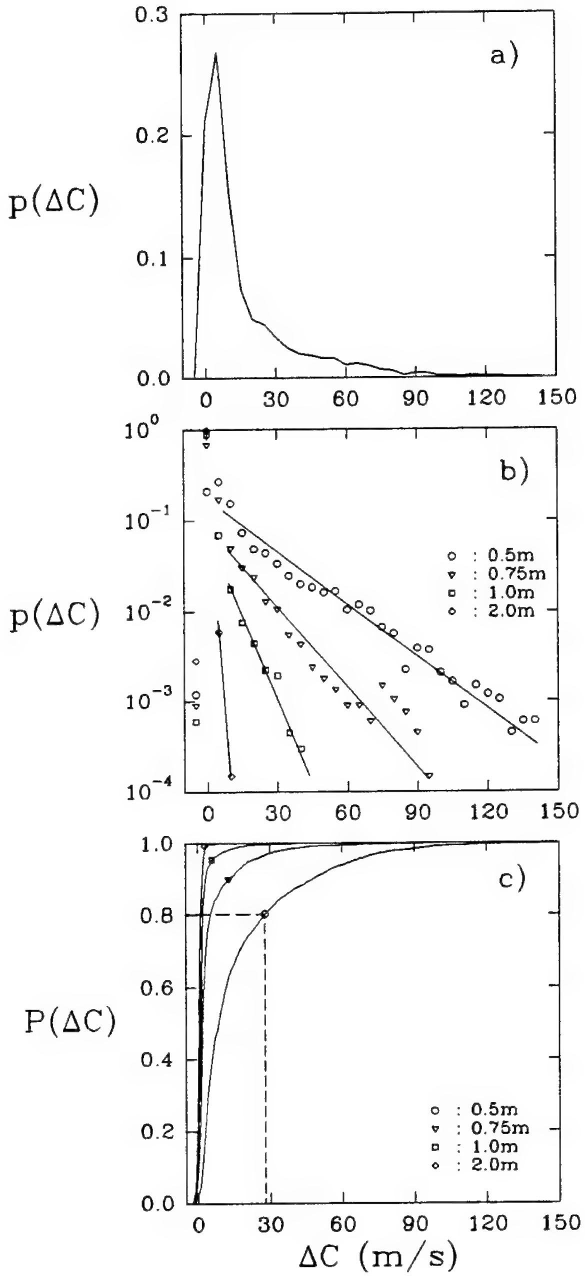


Figure 7.8: a) Probability distribution of the sound-speed anomaly at 0.5m for three 20min record (f=10kHz). The bin size used for the distribution is 5m/s. b) Same as in a) but plotted with three other depths on a log-linear scale. The straight lines show the trend of the data away from the noise region. The distribution at 2.0m is representative of the measurement's noise. c) Cumulative probability distribution corresponding to the distributions in b). The dotted line in c) shows that at a depth of 0.5m, 20% of the data had a sound-speed anomaly greater than 27m/s. Woods Hole data.

The cumulative probability distributions are shown in figure 7.8c for the same four depths. The dotted line on the distribution at 0.5m indicates that 20% of the measurements had anomalies above 27m/s. Thus, although the excursions of the sound-speed anomalies can be very large, the frequency of their occurrence is relatively small.

7.4 Depth dependence of the sound-speed anomalies

The three time-series in figure 7.1 clearly show that the amplitude of the sound-speed anomalies decrease with depth. The averages of six 20min time-series obtained at a wind speed of 7 to 8m/s were computed and plotted as a function of depth in figure 7.9. The data show some scatter but nevertheless they have a distinct trend corresponding to an exponential decrease with depth.

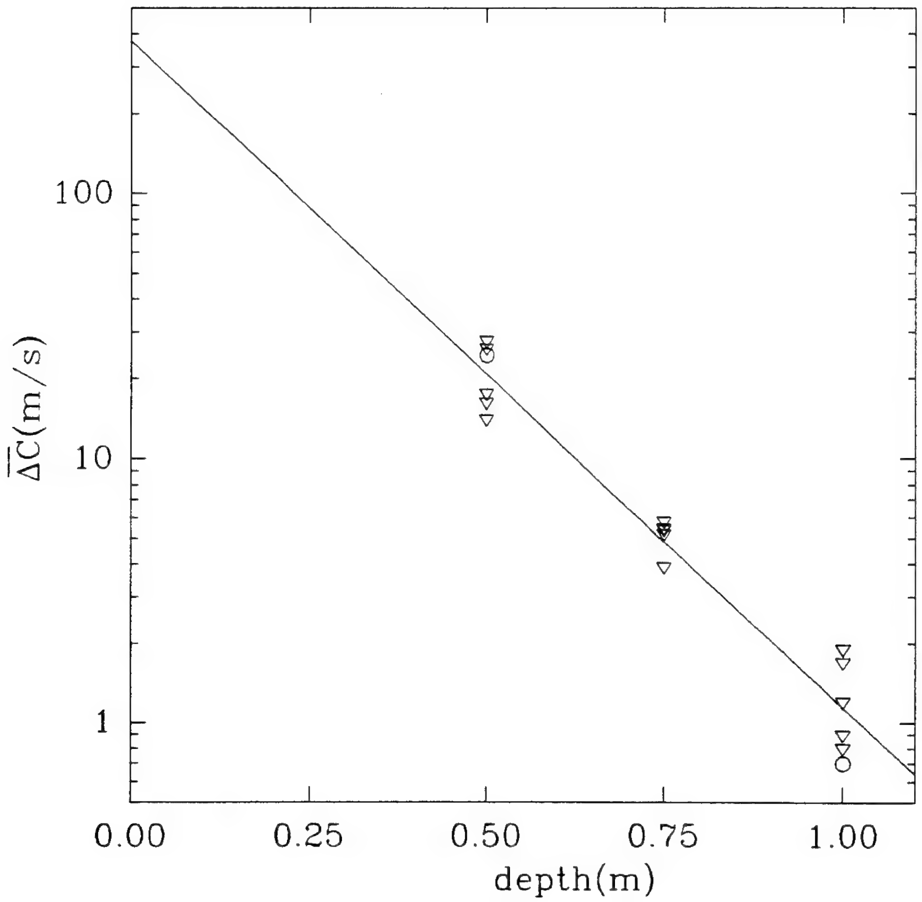


Figure 7.9: Average sound-speed anomaly $\overline{\Delta c}$ as a function of depth. Each data symbol is a 20min average. (∇) Woods Hole experiment and (o) San Diego experiment. The solid line is the best exponential fit to the data. Wind speed of 7 to 8m/s. The frequency of the acoustic pulses was either 5kHz or 10kHz.

The equation describing the exponential is

$$\overline{\Delta c} = 370\text{m/s}\ e^{\frac{-d}{0.18\text{m}}} \qquad (7.2)$$

where d is the depth from the free surface. The e-folding depth of the measurements is 0.18m and the anomaly extrapolated to the surface is 370m/s. For comparison, the exponential decrease described by the measurements of Farmer and Vagle [1989] at a wind speed of 10m/s show an e-folding depth of 1.4m and an anomaly at the surface of 3m/s (refer to figure 1.9). There are approximately one to two orders of magnitude differences for the e-folding depth and the surface anomaly between our measurements and those of Farmer and Vagle [1989]. The sound-speed profile described here is much more pronounced and shallower than the one previously measured by Farmer and Vagle [1989]. We leave to the Discussion further comments on these differences.

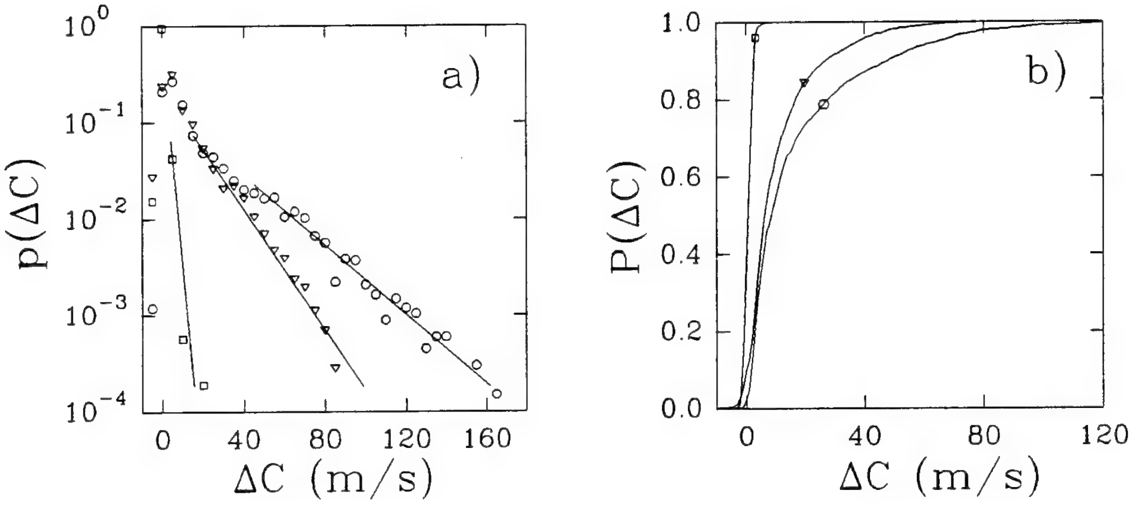


Figure 7.10: a) Probability distribution for three different wind conditions. Each distribution was obtained from three 20min time-series. (o) 7.5 to 8m/s (WH), (∇) 5.5 to 6m/s (SD), (□) less than 3m/s (SD) (representative of the noise level). b) Cumulative probability distribution for the same data.

Although it may be clear that in general the average sound-speed anomaly increases with wind speed, it is less evident how the probability distribution of the signal may change with a change in wind conditions. Three sets of three 20min time-series were selected to cover a significant range of wind speeds. All data selected were for acoustic frequencies of 10kHz or less. The probability distributions of the sound-speed anomalies at a depth of 0.5m for all three data sets (1hr each set) are shown in figure 7.10a along with the cumulative distributions in figure 7.10b. The probability distributions in figure 7.10a show an approximate exponential relationship with a slope that steepens with a decrease in wind speed. An interesting feature of the two probability distributions at the higher wind speeds is their close match up to 40m/s after which the slopes diverge. This suggests that the effect of higher wind speed is primarily to increase the frequency of occurrences of high sound-speed anomalies while keeping relatively constant the frequency of occurrences of smaller anomalies. In contrast, the depth dependency of the probability distributions in figure 7.8a showed a clear decrease in the frequency of occurrences of both low and high anomalies with an increase in depth (along with a change in the slope of the distribution).

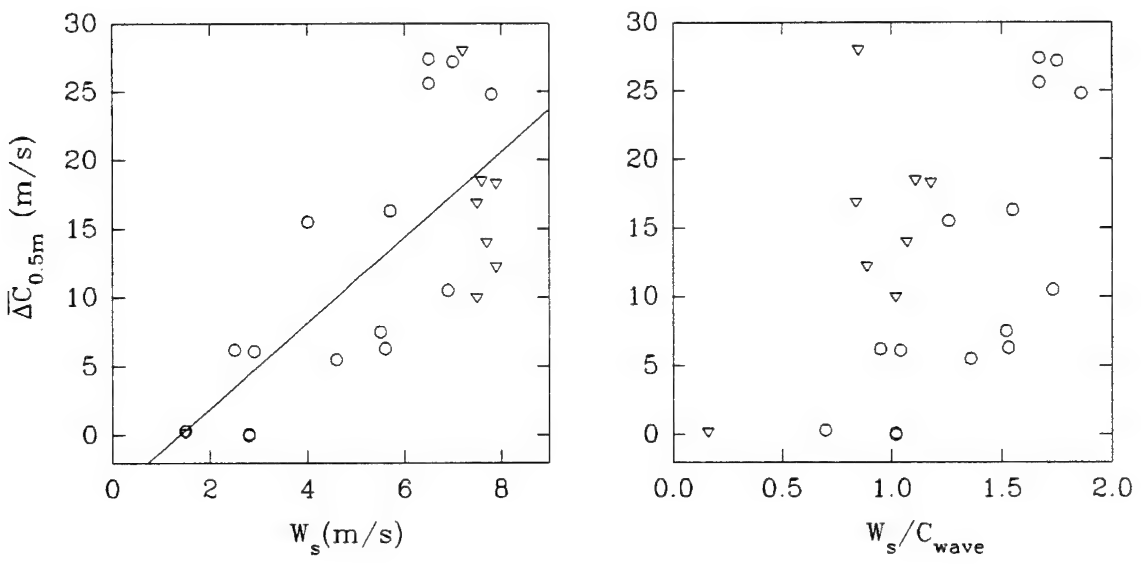


Figure 7.11: Average sound-speed anomaly at 0.5m depth as a function of a) wind speed and b) wind speed normalized by the phase velocity of the dominant waves C_{wave}. Data were averaged over 20min records. Solid line in a) is best linear fit to the data with a 0.74 correlation coefficient. (o) Data from San Diego experiment, (∇) data from Woods Hole experiment.

7.5 Correlation with environmental parameters

Correlation of the average sound-speed anomalies with wind speed and significant wave height were performed. Only the data at 5kHz and 10kHz were selected in order to ensure that the measurements were representative of the low-frequency sound-speed (see §7.7.3). The sound-speed data and the environmental data were averaged over 20min records.

Figure 7.11a shows the average sound-speed anomaly plotted against wind speed and figure 7.11b shows the same data with the wind speed normalized by the phase velocity of the dominant waves[4]. The phase velocity of the dominant waves was obtained from the spectra of the wave gauge measurements for the Woods Hole experiment. For the San Diego experiment, the phase velocity was obtained from a simple wave forecasting equation along with measurements of the wind and knowledge of the fetch in the Santa Cruz Channel [U.S. Army Corps of Engineers, 1984, p3-48].

Although the data in figure 7.11a is correlated, there is nevertheless significant scatter which suggest that an averaging time longer than 20min should be used to improve the correlation. Note that direct extrapolation of the data up to a wind speed of 15m/s, which is often encountered in the open ocean, suggest that the average anomalies could be as high as 40 to 50m/s at a depth of 0.5m. No improvement in the correlation is observed when the wind speed is normalized by the phase velocity of the dominant waves as shown in figure 7.11b.

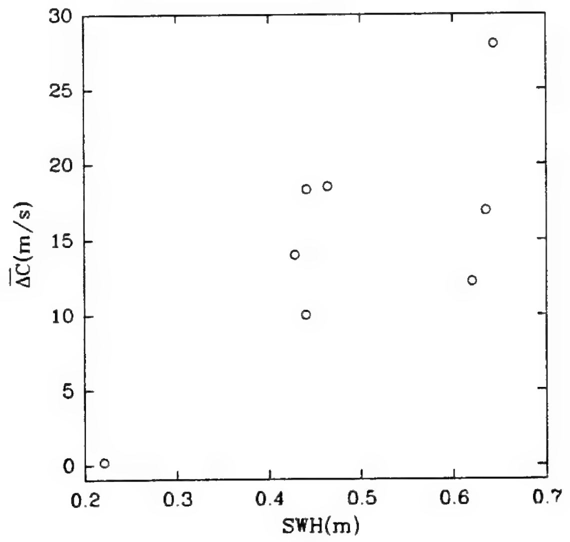


Figure 7.12: Average sound-speed anomaly at a depth of 0.5m as a function of the significant wave height for the Woods Hole data. Data were averaged over 20min records.

[4] W_s/C_{wave} is the reciprocal of the wave age. The wave age is a measure of the stage of development of the wind waves for a given wind speed.

The 20min averaged sound-speed anomaly was also correlated with the significant wave height for the Woods Hole data. The correlation is shown in figure 7.12 were again the limited data set shows significant scatter.

7.6 Highest sound-speed anomaly measured

Figure 7.13 shows a short time-series of the highest sound-speed anomalies measured during the experiments. Notice that the actual sampled data points are shown in order to demonstrate that the high excursions in the signal are not due to single outliers for each event. This particular time period was also recorded by the underwater video camera and two interesting features were viewed on the video recording which help our understanding of the anomalies observed. The underwater camera was equipped with a wide angle lens and it was positioned to look horizontally at a depth of approximately 25cm (details on the camera are given in §5.1 and §5.4). Hence, the camera was capable of observing bubble plumes being formed at the buoy and those being formed some distance away as long as they remained in the field of view.

The vertical arrow in figure 7.13a corresponds to the beginning of a breaking wave on the video recording. This particular wave broke right above the buoy and it injected a large plume of bubbles which were observed to reach the uppermost sensor (the deeper sensors were not in the field of view). Accordingly, large sound-speed anomalies up to 800m/s were measured immediately after the beginning of breaking in figure 7.13. Less than a second prior to this breaking event, another large breaking wave was observed a few meters ahead of the buoy. It is not clear whether these two separate breaking events were from two different waves breaking almost simultaneously or whether they were from the same wave crest breaking at two different locations along the crest. In any case, the important observation to retain is that a second bubble plume was formed some distance away from the buoy and its distinct whiter cloud on the video recording was clearly seen to drift toward the buoy. Approximately 25s later, this second bubble cloud eventually drifted by the buoy and at that time, an appreciably higher bubble density was observed on the video. Concurrently, the sound-speed anomaly increased significantly during the passage of this second cloud. Thus, it would appear that even relatively 'older' clouds which have had plenty of time to degas still contain air concentrations high enough to cause anomalies up to 400m/s.

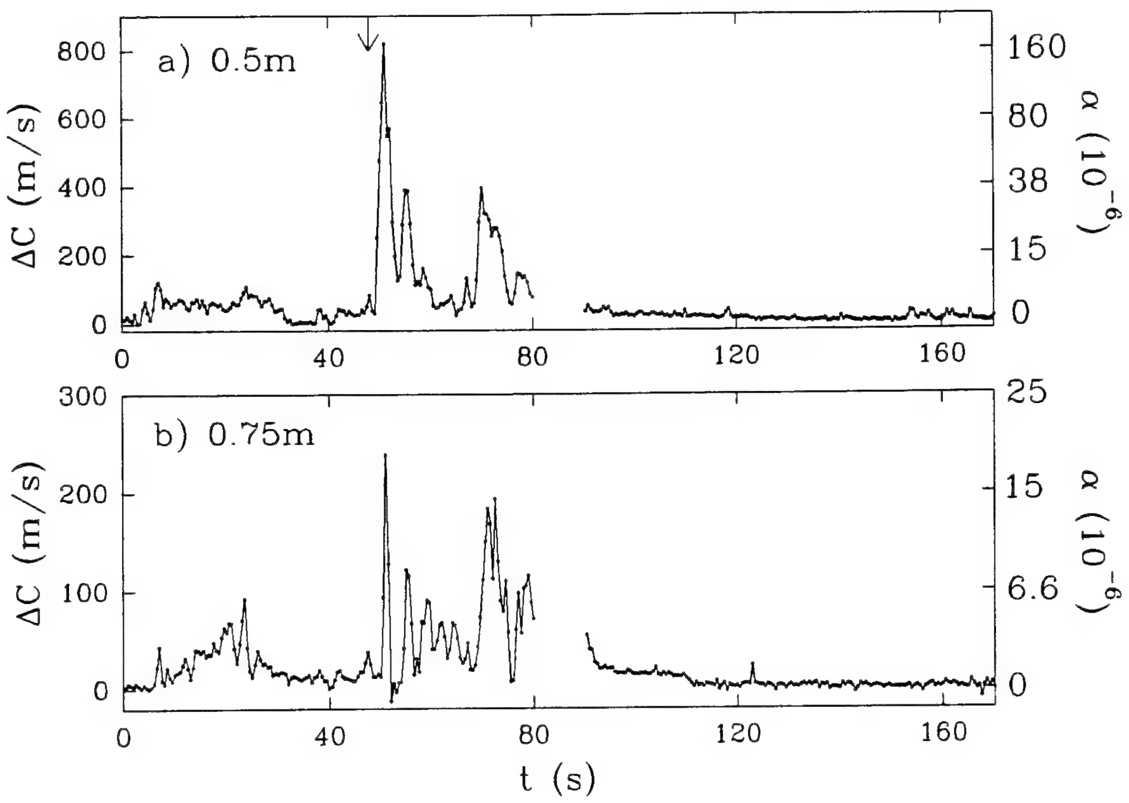


Figure 7.13: Highest sound-speed anomalies measured during both experiments at a)0.5m and b)0.75m. The first large peak at approximately t=50s was caused by a wave which broke at the buoy at the time indicated by the arrow. The second peak approximately 25s later was caused by a bubble cloud formed 25s earlier which drifted by the buoy. The void-fraction corresponding to an 800m/s anomaly is shown on the left axis. The frequency of the acoustic pulses was 5kHz. Woods Hole data.

7.7 Dependence of the sound-speed anomaly and attenuation on the acoustic frequency

Because of the high variability of the sound-speed measurements, it is difficult to adequately compare measurements made at different frequencies unless the measurements are simultaneous. Consequently, a broad-band pulse was used during the Woods Hole experiment to measure the sound-speed and the attenuation at various frequencies simultaneously.

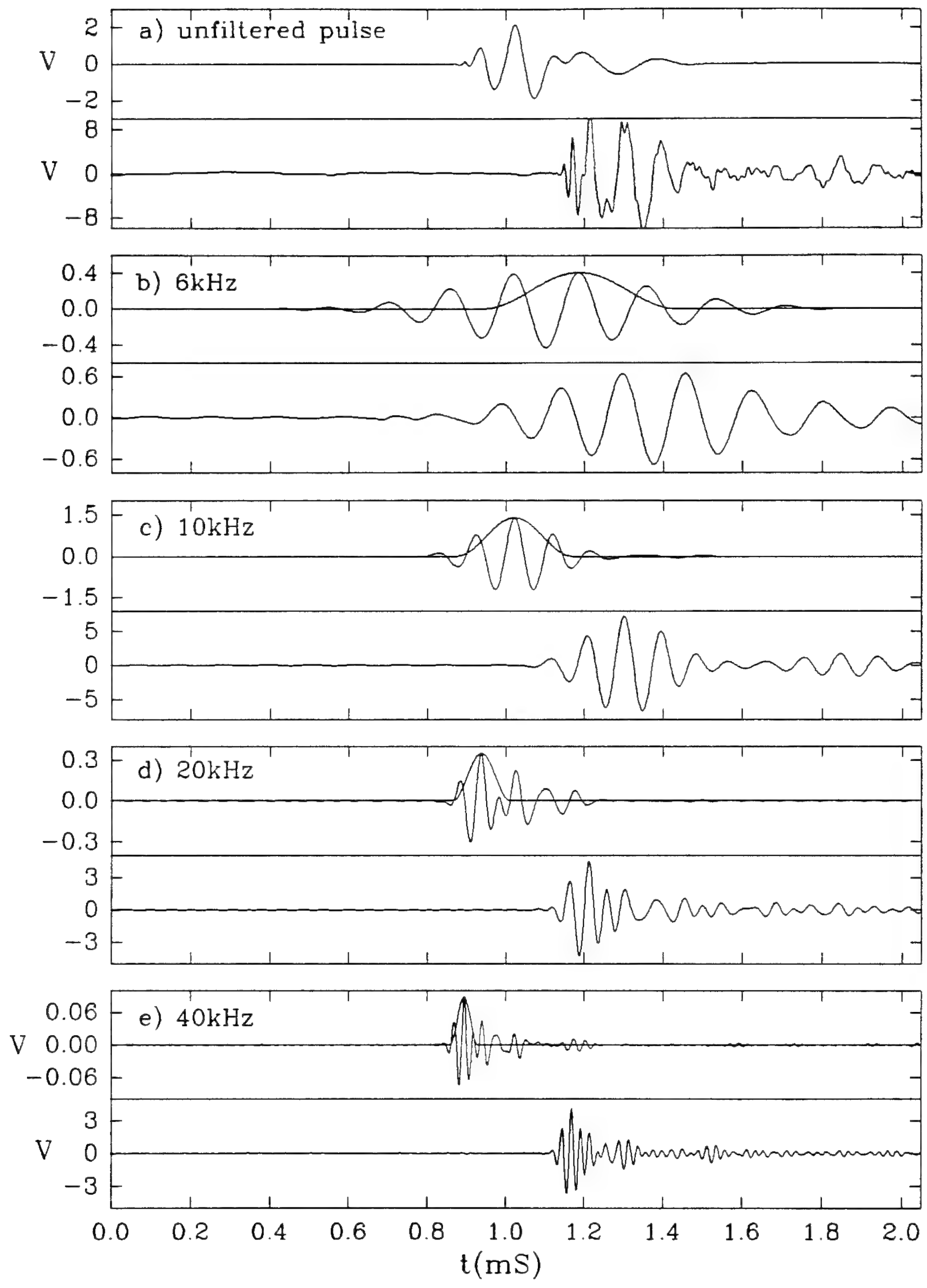


Figure 7.14: a) Transmit (above) and received (below) composite pulses. The filtered version of the composite pulse shown in a) with band-pass filter center frequencies b)6kHz, c)10kHz, d)20kHz and e)40kHz. The window function plotted with each transmit pulses was used to window the signal and thus eliminate the tail-end of the transmit pulses from biasing the cross-correlation function. Woods Hole data.

7.7.1 The composite pulse

The 'composite' pulse is the name given to the broad-band pulse that was synthesized from four different pulses of frequencies 5, 10, 20 and 40kHz. The amplitudes of the four individual pulses were adjusted to compensate for the 12dB/octave transmitter response. The composite pulse was obtained by superposing the four single frequency pulses. Figure 7.14a shows the transmit and received composite pulses. Figure 7.14b,c,d,e show the digitally band-pass filtered version of the pulses shown in figure 7.14a and the band-pass filter center frequencies are indicated on each plot. The characteristics of the band-pass filters are described below. Notice the window function plotted with each transmit signal. This window was applied at the peak of each transmit signal in order to eliminate the tail-end of the received signal from biasing the cross-correlation function. As discussed in §6, the tail-end of the received signal contains unwanted reflections of the acoustic pulse from the water surface and the structural members of the buoy.

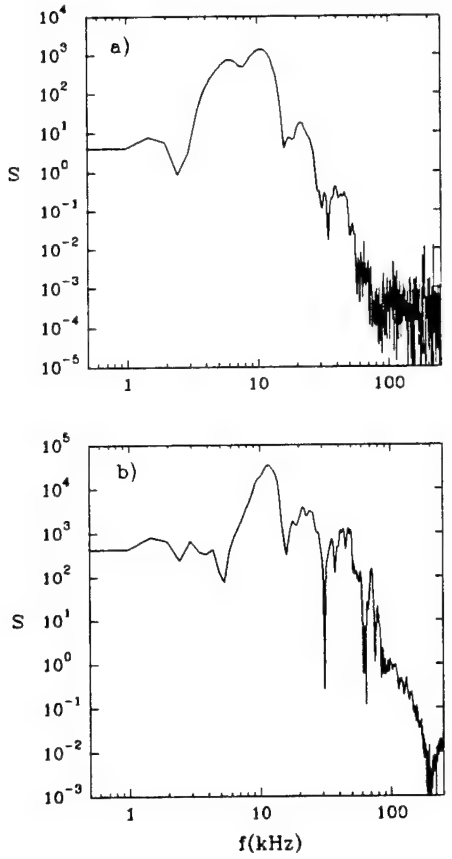


Figure 7.15: Spectrum of a) the transmit composite pulse and b) the received composite pulse shown in figure 7.14a. Note the specific peaks in a) at 5, 10, 20 and 40kHz. Same peaks are found in b) except at 5kHz where the transmit hydrophone did not put out enough power.

The frequency spectra of the transmitted and received pulses shown in figure 7.14a are shown in figure 7.15. The spectra show distinct peaks at 5, 10, 20 and 40kHz except for the received spectrum (figure 7.15b) which does not show a clear peak at 5kHz. This indicates that the transmit hydrophone did not put out enough power at that frequency and that more energy should have been placed in the 5kHz band of the transmit signal. Consequently, the S/N ratio of the filtered signal at 5kHz was relatively low and it was therefore decided to use a slightly higher frequency (6kHz) for the band-pass filtering.

The digital band-pass filtering of the composite pulses was performed after the Woods Hole experiment was completed. The data were band-pass filtered with finite impulse response (FIR) filters. The characteristics of the various FIR filters used are given in table 7.1. Because of the broad-band nature of the transmit and received pulses, it was possible to filter the composite pulse at frequencies different from the four principal frequencies (i.e. 5, 10, 20 and 40kHz). It should also be pointed out that the results presented in this section are insensitive to changes in the shape of the band-pass filters as long as the filter characteristics remained reasonable (i.e. same center frequency and a relatively tight band-pass region).

f_c (kHz)	f_{low} (kHz)	f_{high} (kHz)	N
6	5.5	6.5	512
7.5	6.5	8.5	512
10	7.5	12.5	256
15	13.75	18.75	256
20	15	25	128
25	22.5	27.5	128
30	28.75	33.75	128
35	32.5	37.5	128
40	30	50	64

Table 7.1: Characteristics of the digital FIR band-pass filters used to filter the composite pulses. The 3dB points of the band-pass filter are located at f_{low} and f_{high} and the center frequency is f_c. The number of points in the filter is N.

7.7.2 The technique for measuring the attenuation of the acoustic pulses

The technique for extracting the amplitude information and ultimately the attenuation from the acoustic pulses is shown in figure 7.16. First, the time-delay τ is computed from the cross-correlation. The closest peak located at a time-interval τ away from the dominant peak of the transmit signal is located next (labeled 1). Then, the closest trough ahead of the peak labeled 1 is finally located (labeled 2). The amplitude between the trough and the peak characterizes the amplitude of the pulse. The attenuation is given by the ratio of the amplitude measured in bubbly water (rough sea-state) to that of the amplitude measured in bubble-free water (calm sea-state).

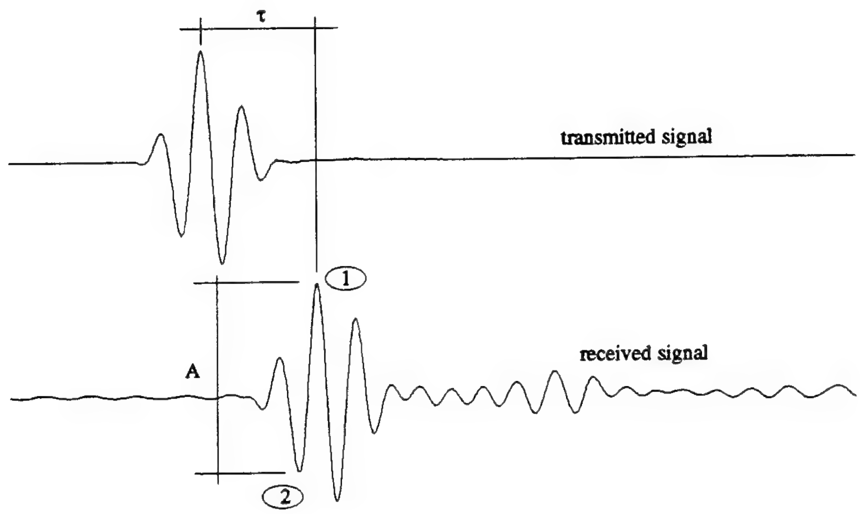


Figure 7.16: Schematic of the measurement of the pulse amplitude. Transmit pulse above and received pulse below. First, the time-delay τ is computed from the cross-correlation. The closest peak located at a time-interval τ away from the dominant peak in the transmit signal is located next (labeled 1). Then, the closest trough ahead of the peak labeled 1 is finally located (labeled 2). The amplitude between the trough and the peak characterizes the amplitude of the pulse.

7.7.3 Measurements of the frequency dependent sound-speed and attenuation

Figure 7.17 shows time-series of the sound-speed anomaly obtained by band-pass filtering a 20min record of composite pulse data at various frequencies from 6 to 40kHz. After band-pass filtering, the pulses are cross-correlated in the usual manner to obtain the time-delay and hence the sound-speed anomaly. The signal in figure 7.17 shows the characteristic large intermittent fluctuations which are caused by bubble clouds drifting by the sensors. The signal is highly correlated and of similar magnitude up to approximately 20kHz. Beyond 20kHz, the signal is still correlated but its magnitude changes drastically.

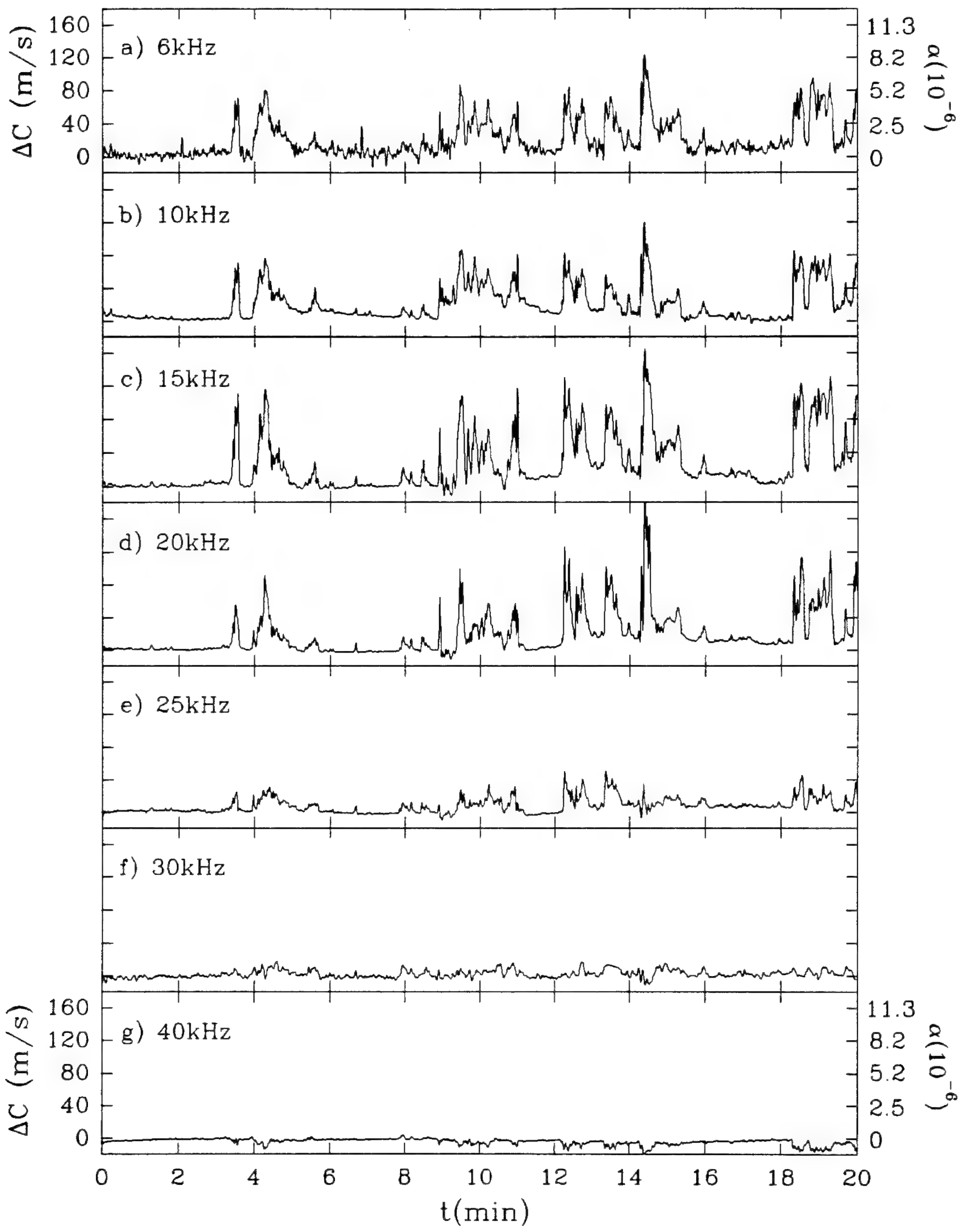


Figure 7.17: Sound-speed anomalies at various frequencies obtained by band-pass filtering a 20min record of composite pulse data acquired at a depth of 0.5m. The various band-pass frequencies are indicated on the plots. The axes on the right-hand side show the corresponding void-fraction. The wind speed and the SWH were 8m/s and 0.46m respectively. Woods Hole data.

At 40kHz, the sound-speed anomalies become negative and thus the sound-speed slightly increases inside the bubble clouds. This behavior has been observed in the past at a similar frequency and it is believed to be caused by local peaks in the bubble population or changes in its slope [Medwin, 1974; Medwin et al. 1975a; also see figure 1.11]. These local inhomogeneities in the bubble population can produce significant changes in the sound-speed at frequencies where bubbles can resonate and thus generate important phase shifts.

Below 20kHz, the fluctuations are caused by the void-fraction in the water and not by the characteristics of the bubble size distribution. Hence, the sound-speed in that range in not a function of frequency. Above 20kHz, the sound-speed reductions caused by the void-fraction are still present but the effect of local features in the bubble distribution function start to alter the character of the signal and these effects become more important as the frequency increases. However, the features in the signal remain correlated at all frequencies. This was anticipated since the sound-speed fluctuations caused either by the void-fraction or by bubble induced dispersion must always coincide with the presence of a bubble cloud at the sensors.

The signal attenuation can also be computed from the variations in the amplitude of the received acoustic pulses with respect to the signal amplitude in bubble-free water which is obtained at low sea-state. After band-pass filtering the composite pulses, the attenuation is computed as outlined in §7.7.2. Figure 7.18 shows time-series of the attenuation for the same data record as the sound-speed anomaly measurements shown in figure 7.17. The similarities between the two figures are striking. The high-frequency attenuation (say above 20kHz) complements the sound-speed anomalies at low-frequencies (say below 20kHz). At high frequencies, the acoustic pulses start to excite bubbles into resonance and this causes increased attenuation of the signal up to 40dB/m in the time-series shown. This effect has been observed by many investigators studying bubbly flows and a good review of measurements and theory can be found in Commander and Prosperetti [1989].

At 6kHz, the attenuation measurements are rather small but there appear to be some features which are correlated with the presence of bubble clouds. However, these fluctuations have small negative attenuation suggesting that the signal increases slightly in amplitude while inside the bubble clouds. The reason for this is still unclear since theory always predicts an increase in attenuation in the presence of bubbles at any frequency.

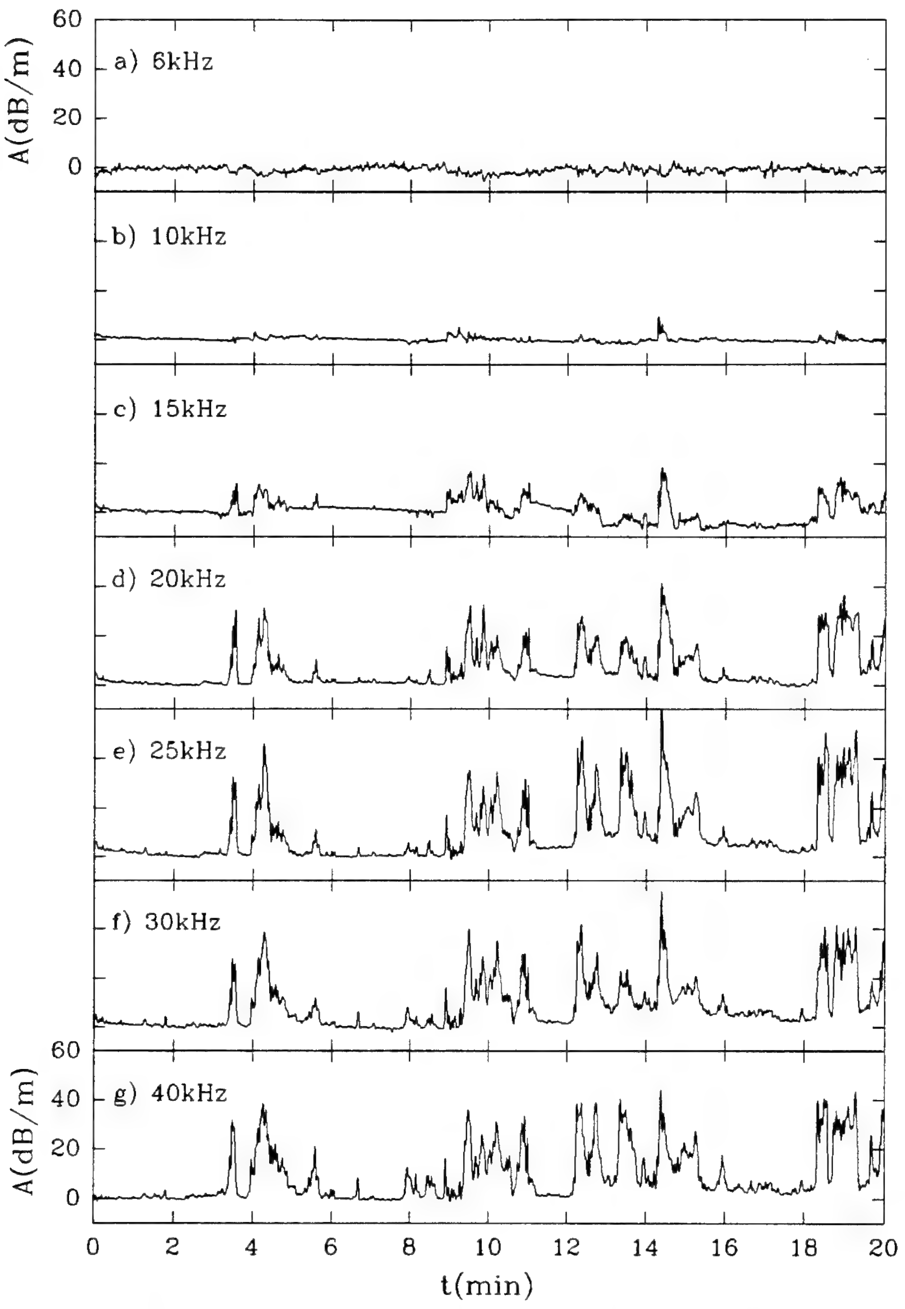


Figure 7.18: Attenuation at various frequencies obtained by band-pass filtering a 20min record of composite pulse data acquired at a depth of 0.5m. The various band-pass frequencies are indicated on the plots. The wind speed and the SWH were 8m/s and 0.46m respectively. Woods Hole data.

The average of the sound-speed time-series shown in figure 7.17 and attenuation time-series shown in figure 7.18 were computed at all frequencies and the results are shown in figure 7.19. As expected from the previous analysis of the time-series, the average sound-speed anomaly $\overline{\Delta c}$ is almost independent of frequency at frequencies below 20kHz. Thus the data show that the low-frequency asymptotic limit which occurs when the sound-speed is solely dependent on the void-fraction is reached at approximately 20kHz at a wind speed of ~8m/s. However, it is likely that this limit will be a function of the sea-state and that at higher sea-states the limit may be at a lower frequency since more larger bubbles are likely to be present below the surface in these conditions. At frequencies above 20kHz, the average sound-speed anomaly decreases rapidly and it becomes slightly negative at 35 and 40kHz. The attenuation (figure 7.19b) shows an approximately steady increase up to 25kHz and it levels off past 25kHz.

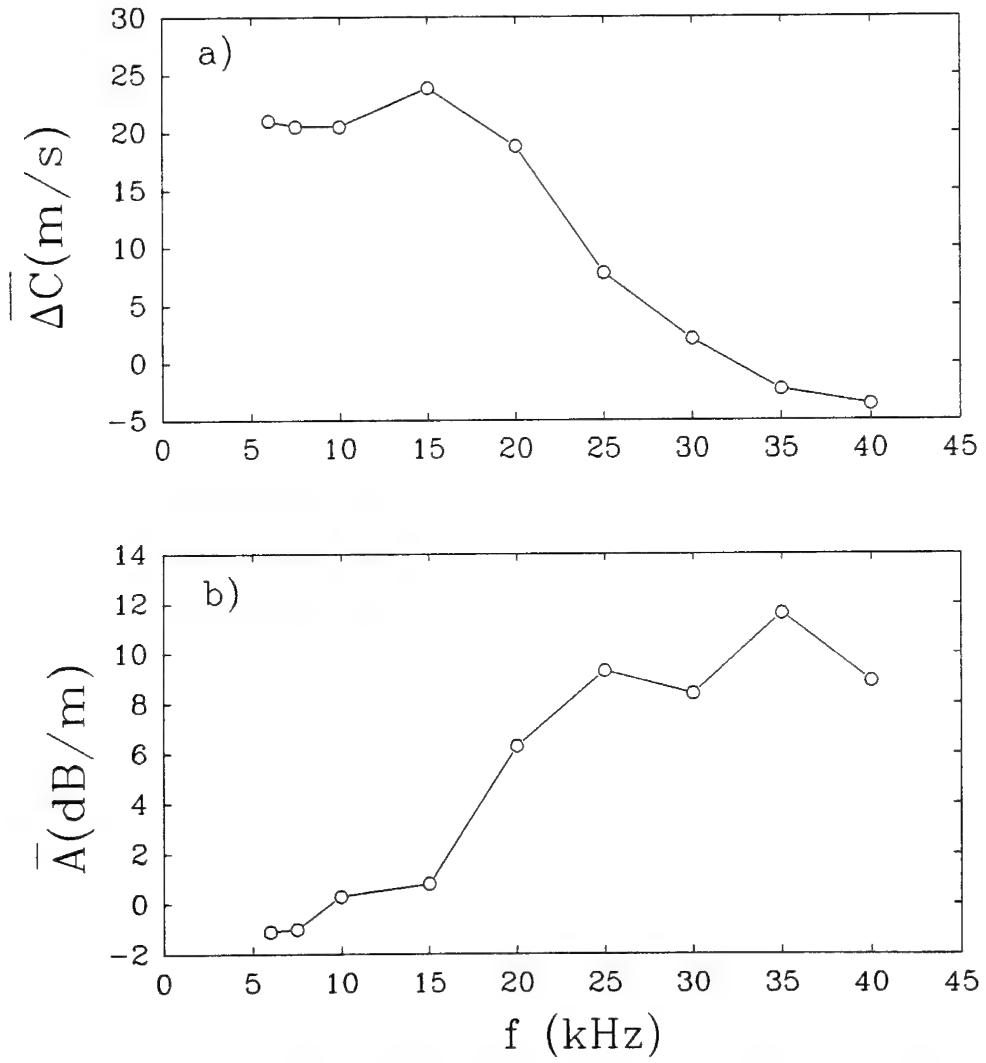


Figure 7.19: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency. The averages were computed over the 20min time-series shown in figure 7.17 and figure 7.18. Note how $\overline{\Delta c}$ is almost independent of frequency below 20kHz.

Figure 7.20 shows the standard deviations of the sound-speed anomaly $\sigma_{\Delta c}$ and attenuation σ_A. It is interesting to notice that the magnitude of the standard deviations are comparable to the averages shown in figure 7.19. This emphasizes the importance of the short-term fluctuations and it shows that characterization of the sound-speed and attenuation signals only in terms of averages may not always be appropriate. Note that the peak in $\sigma_{\Delta c}$ at 15kHz also corresponds to an increase in $\overline{\Delta c}$ (figure 7.19a) at the same frequency. The reason for this may be that signal fluctuations increase at the transition zone where dispersive effects begin.

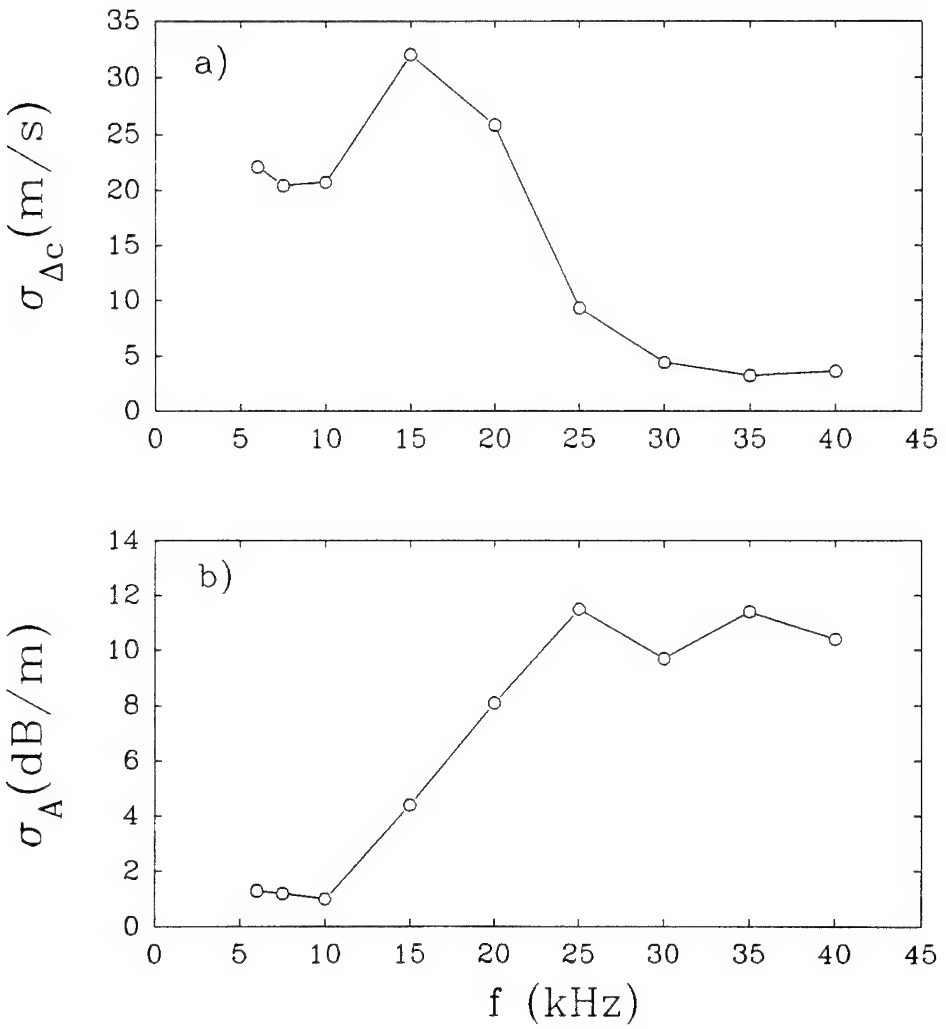


Figure 7.20: Standard deviation of a) the sound-speed anomaly $\sigma_{\Delta c}$ and b) attenuation σ_A as a function of frequency. The standard deviations were computed over the 20min time-series shown in figure 7.17 and figure 7.18.

All of the results presented so far on the dependency of the sound-speed and attenuation on frequency involved data acquired at a depth of 0.5m. Figure 7.21 shows time-series of the depth dependency for the same data record at three different depths. The 10kHz sound-speed anomalies for three different depths are compared with the 40kHz attenuation measurements for the same three depths. The high-frequency attenuation closely follows the low-frequency sound-speed anomaly at all depths. As pointed out earlier in §7.3 and §7.4, the sound-speed anomalies decrease very rapidly with depth and this is the case again in figure 7.21a,b,c. Figure 7.21d,e,f show that this is also true for the attenuation at high-frequencies.

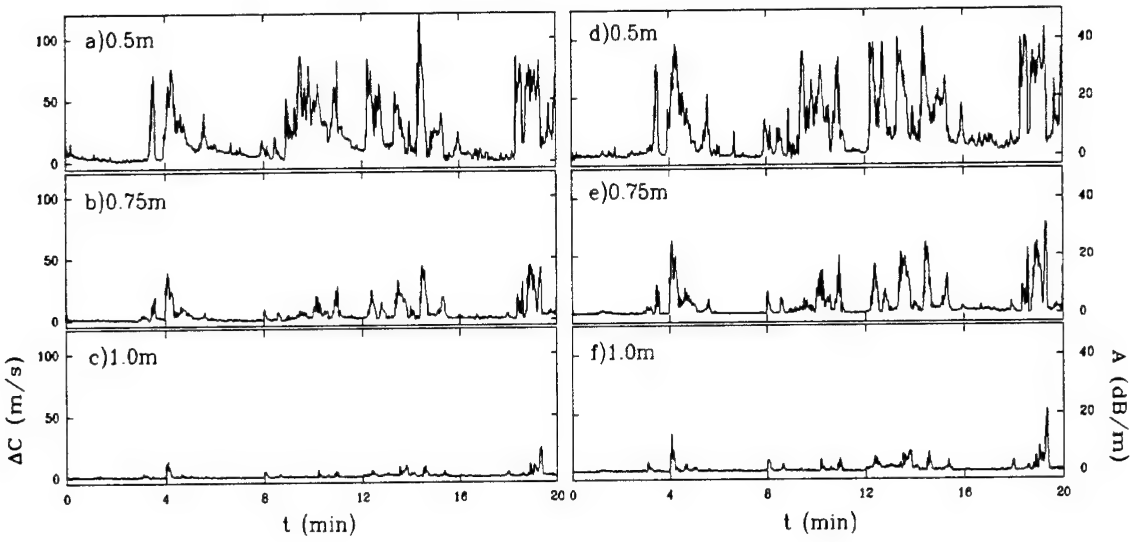


Figure 7.21: Sound-speed anomaly for 10kHz acoustic pulses at a depth of a)0.5m, b)0.75m and c)1.0. Attenuation for 40kHz acoustic pulses at a depth of d)0.5m, e)0.75m and f)1.0m. Same data record as the one used in figure 7.17 and figure 7.18.

7.7.4 Cumulative probability distributions as a function of frequency

Although the magnitude of the averages and standard deviations of the sound-speed anomalies were shown to be similar at low frequencies (10kHz and below), it is interesting to compare their probability distributions in order to verify the similarity of the signal structure. Figure 7.22a shows the cumulative probability distributions of the sound-speed anomalies at 0.5m for the frequencies 6, 7.5 and 10kHz. Notice that all three distributions are very similar indicating that the structure of the signal at low frequencies are also similar. By contrast, figure 7.22b shows that the distributions are very different if the selected frequencies extend over to the transition zone (around 15 to 25kHz) and into the higher frequency region (above 25kHz).

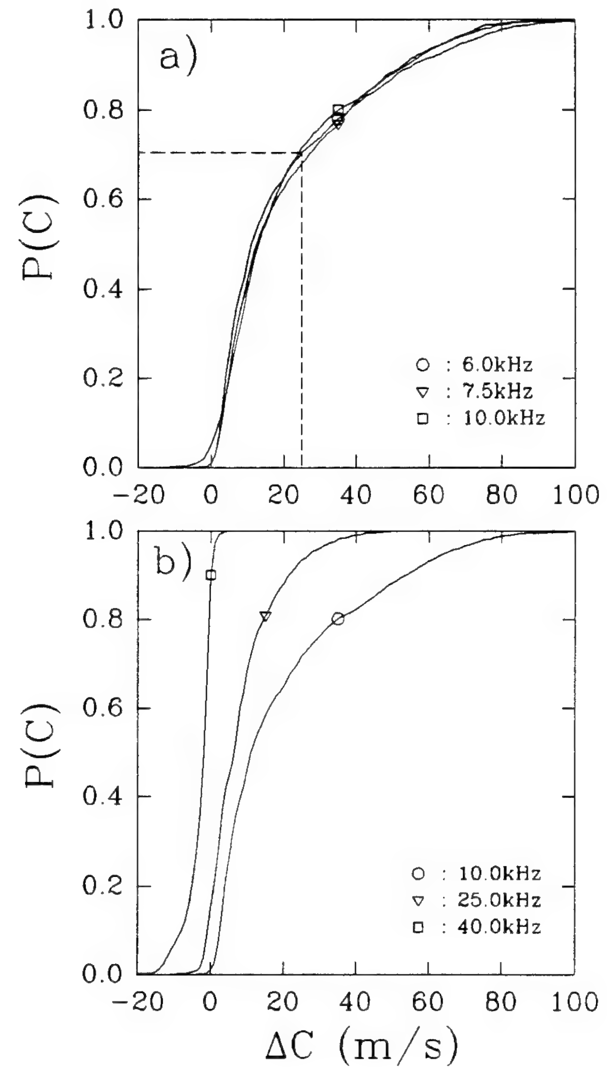


Figure 7.22: Cumulative probability distributions of the sound-speed anomaly at a depth of 0.5m for a) three low frequencies and b) three different frequencies spanning the range of the measurements. The dotted line shows that 30% of the time the signal was above 24m/s. Same data record as the one used in figure 7.17 and figure 7.18.

The same observations can be formulated for the cumulative distributions of the attenuation at high-frequencies (figure 7.23a) where the distributions are relatively similar. For a broader range of frequencies extending down to the low frequency region (say 10kHz and below), the cumulative distribution of the attenuation shown in figure 7.23b are very different as expected from the character of the time-series in figure 7.18.

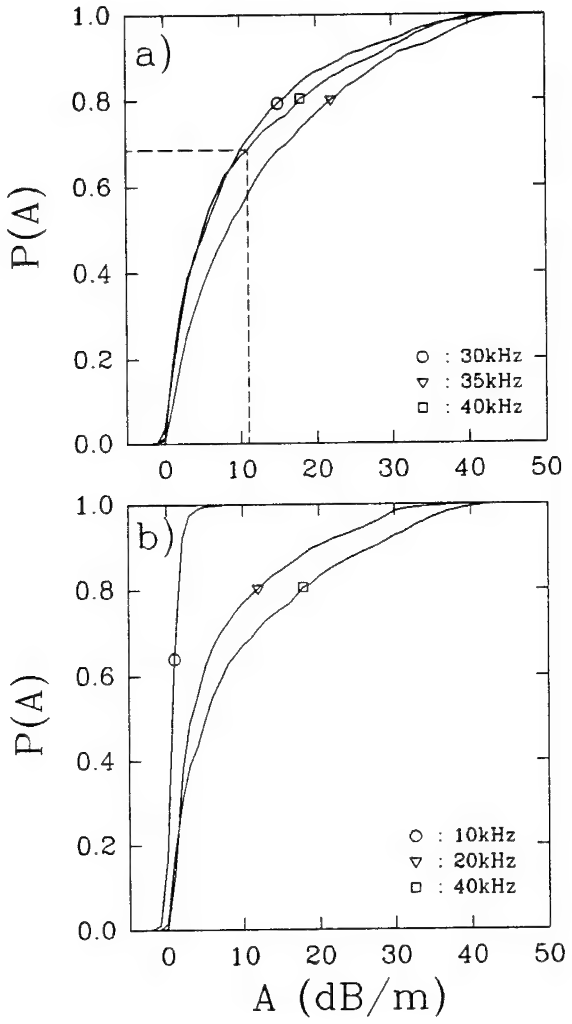


Figure 7.23: Cumulative probability distributions of the attenuation at a depth of 0.5m for a) three high frequencies and b) three different frequencies spanning the range of the measurements. The dotted line shows that 30% of the time the signal was attenuated by more than 12dB/m. Same data record as the one used in figure 7.17 and figure 7.18.

7.8 Bubble population information from sound-speed and attenuation measurements

It is possible to infer the attenuation and propagation speed of an acoustic wave in bubbly water by integrating the effects of each bubble over the full bubble population spectrum. This is usually referred to as the 'forward' problem and the two independent integral equations necessary to solve it are given in appendix B (equation B.18 and B.19). The only input for the forward problem is the bubble population. This is essentially the technique which has been used by Farmer and Vagle [1989] to infer the near-surface sound-speed from sonar derived bubble population measurements.

The 'inverse' problem requires solving the same integral equations to obtain the bubble population from measurements of the attenuation and propagation speed. The frequency-dependent measurements of the complex propagation velocity presented in the last section will be used to infer the characteristics of the bubble population. The integral equations to be solved are independent and they either describe the real (propagation speed) or the imaginary (attenuation) part of the complex propagation velocity. Thus, they give two independent measures of the bubble population.

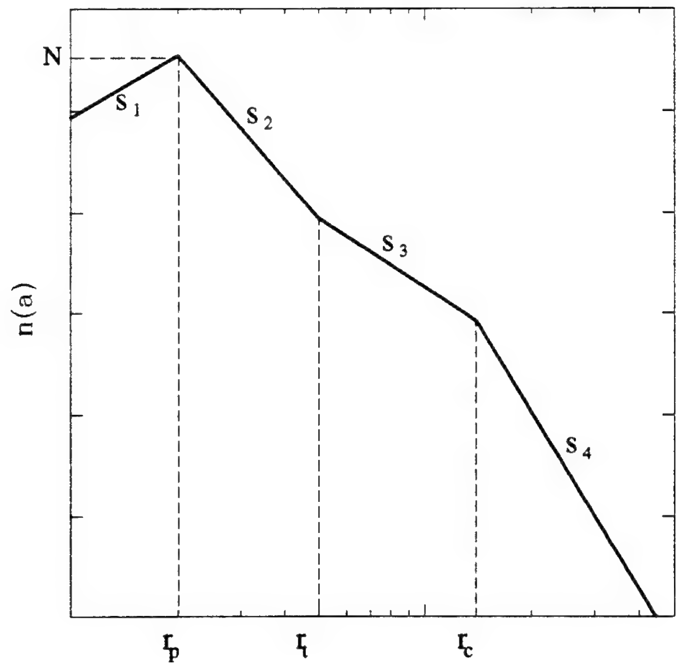


Figure 7.24: A conceptual model of the bubble population showing various slopes and critical radii which define the shape of the population. The number of bubbles per m^3 per μm increment is given by n(a) and N is that number at the peak. The subscripts 'p', 't' and 'c' stand for peak, transition and cutoff

The solution of the inverse problem presents some computational difficulties and recent studies have started to address this issue with elaborate finite element solutions [Commander and McDonald, 1991]. The implementation of such an elaborate computational scheme is beyond the scope of this thesis. Instead, it was decided to implement a simpler 'forward' technique where the solutions for a full range of bubble population parameters are computed and the variance of the solutions with respect to the attenuation and sound-speed measurements is minimized in order to obtain the bubble population parameters which best fit the measurements.

In order to reduce the computational time required for such a direct approach, a realistic range of bubble population parameters was defined according to previous studies of bubble population at sea. A brief review of these studies was given in §1.4. In figure 7.24 we show a conceptual model of the bubble population based on these previous investigations. Many different slopes and critical radii have been included to permit the definition of the most realistic population possible.

For the eight variable parameters shown in figure 7.24, we define in table 7.2 the ranges of values which will be surveyed in the computation of the forward problem. Every combination is attempted yielding approximately 250 000 different solutions for the attenuation and the propagation velocity. Notice that only one option is given for the parameters s_1, s_2 and r_p. The actual values of the parameters s_1 and r_p have little importance in the following analysis because they affect mostly the very high frequency acoustic waves, say above 100kHz, and all of the measurements presented here are limited to frequencies of 40kHz and below. Furthermore, it will be shown later that the contribution of the very small bubbles to the void-fraction is negligible. The value of $s_2 = -4$ over the range of radii from 20μm up to some transition radii r_t has received substantial support in the bubble population literature. For the other parameters, there are still discrepancies in the literature and the range shown in table 7.2 covers most of the previously reported results.

Of critical importance in the estimation of the contribution of large bubbles to the void-fraction (and thus the sound-speed) and to gas transfer is the character of the bubble population at large bubble radii (say greater than 100μm). Few field studies have addressed this issue which is very important when the slope of the population of the larger bubbles is close to -3 (refer to §1.4 for a more detail exposition on this issue). Thus, we are particularly interested in determining the values of s_4 and r_c.

N	s_1	s_2	s_3	s_4	r_p (μm)	r_t (μm)	r_c (μm)
100000	2	-4	-1.5	-3.0	20	25	80
150000			-2.0	-3.5		30	100
200000			-2.5	-4.0		40	120
250000			-3.0	-4.5		50	140
300000			-3.5	-5.0		60	160
350000			-4.0	-5.5		70	180
400000			-4.5	-6.0		80	200
450000			-5.0	-6.5		90	220
500000				-7.0		100	250
550000				-8.0		110	300
600000				-9.0		120	350
650000				-10.0		130	
700000							
750000							
800000							
850000							
900000							
950000							
1000000							

Table 7.2: Range of parameters surveyed in the search of the best fit solution to the measurements of the attenuation and propagation speed.

Since both the attenuation and the propagation velocity yield an estimate of the bubble population, the complex sound-speed anomaly is computed in order to optimize both the real and the imaginary part of the complex propagation velocity. The complex anomaly is given by

$$\Delta c = c_w - c \tag{7.3}$$

where Δc is now considered to be complex and c is given by

$$c = \frac{\omega}{k_{re} - i\, k_{im}} \tag{7.4}$$

and ω is the real radian frequency of the acoustic wave. The real wavenumber k_{re} is related to the propagation velocity through $c_{re} = \omega k_{re}/(k_{re}^2 + k_{im}^2)$ and the imaginary wavenumber k_{im} is the attenuation of the pressure wave (see appendix B for details). Note that the sign of the imaginary wavenumber k_{im} is negative according to the convention used in appendix B. Substituting equation 7.4 into equation 7.3 and taking the magnitude of the complex anomaly we obtain

$$|\Delta c| = \left(\left(c_w - \frac{\omega\, k_{re}}{k_{re}^2 + k_{im}^2}\right)^2 + \left(\frac{\omega\, k_{im}}{k_{re}^2 + k_{im}^2}\right)^2\right)^{1/2}. \qquad (7.5)$$

It should be pointed out that for the present applications $k_{im} << k_{re}$ and consequently the phase velocity can be approximated as $c_{re} = \omega/k_{re}$. For example, table 7.3 shows the values for k_{re} and k_{im} for the data shown in figure 7.19.

f (kHz)	Δc_{re} (m/s)	A (dB/m)	k_{re} (m^{-1})	k_{im} (m^{-1})	$\lvert\Delta c\rvert$ (m/s)
6	21	-1.1	25.4896	-0.1266	22.3
10	20.5	0.3	42.4683	0.0345	20.5
15	23.8	0.8	63.8449	0.0921	23.9
20	18.8	6.3	84.8391	0.7253	22.8
25	7.8	9.3	105.2671	1.0707	17.1
30	2.1	8.4	125.8399	0.9671	11.7
35	-2.3	11.6	146.3832	1.3355	13.9
40	-3.6	8.9	167.1504	1.0247	9.9

Table 7.3: Values of k_{re} and k_{im} for the data shown in figure 7.19. Note how $k_{im} << k_{re}$. The values for $|\Delta c|$ are given by equation 7.5.

The objective of the computational survey is to minimize the variance between the data and the bubble population model over all frequencies where measurements are available. The variance is defined as

$$\sigma^2 = \frac{\sum_f (|\Delta c|_c - |\Delta c|_m)^2}{N} \qquad (7.6)$$

where $|\Delta c|_c$ and $|\Delta c|_m$ are the computed and measured magnitudes of the complex sound-speed anomalies and N is the number of frequencies where measurements are available. The combination of bubble population parameters which minimizes this summation will yield the closest fit to the combined attenuation and sound-speed measurements obtained as a function of frequency (these measurements are shown in figure 7.19 and given in table 7.3).

The results of the computational survey yielded the following parameters as being the optimal bubble population fitting both the attenuation and the sound-speed measurements (table 7.4)

N	s_1	s_2	s_3	s_4	r_p (μm)	r_t (μm)	r_c (μm)
550000	2	-4	-3.0	-6.0	20	60	160

Table 7.4: Optimal bubble population parameters obtained from the computational survey of all parameter combinations given in table 7.2.

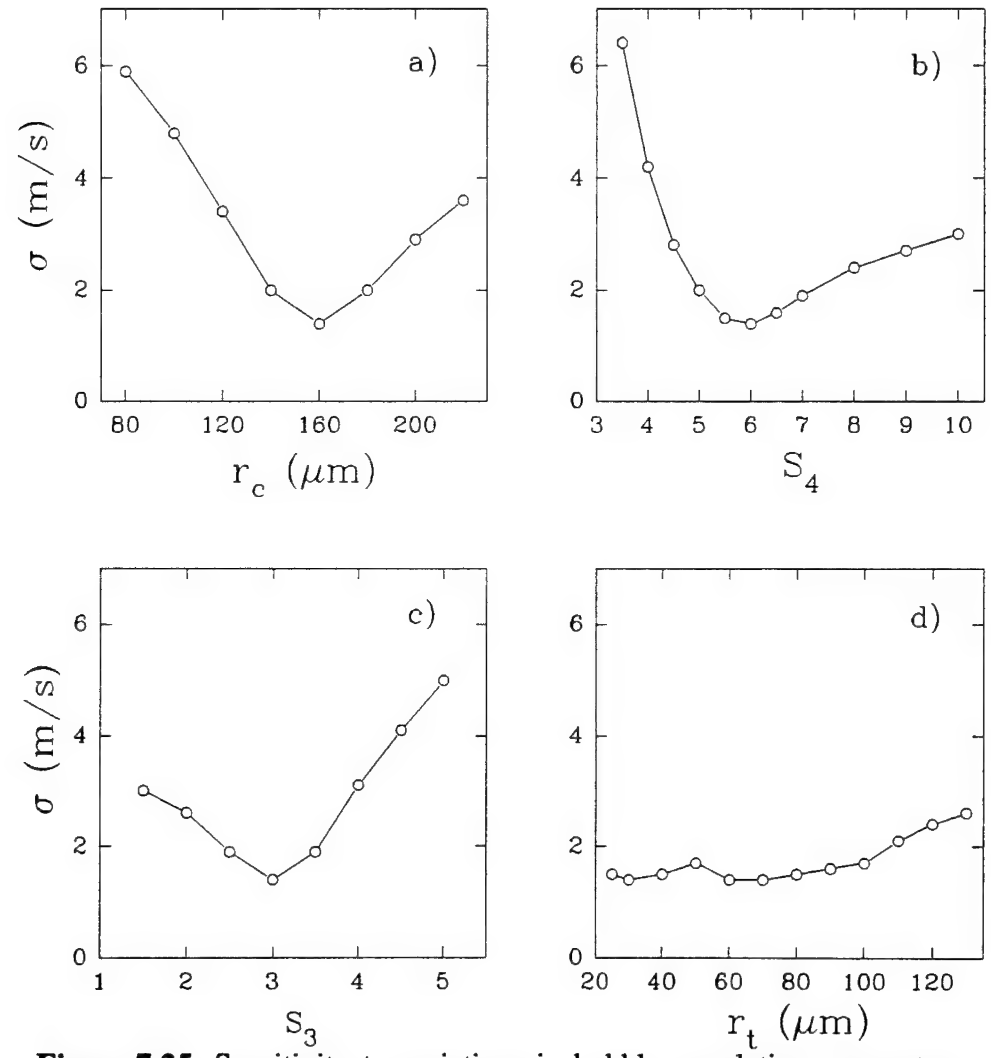


Figure 7.25: Sensitivity to variations in bubble population parameters for a) r_c, b) s_4, c) s_3 and d) r_t. The standard deviation σ between the model and the measurements is given by equation 7.6. In each plot, the default bubble population used is given in table 7.4, and one of its parameters is varied.

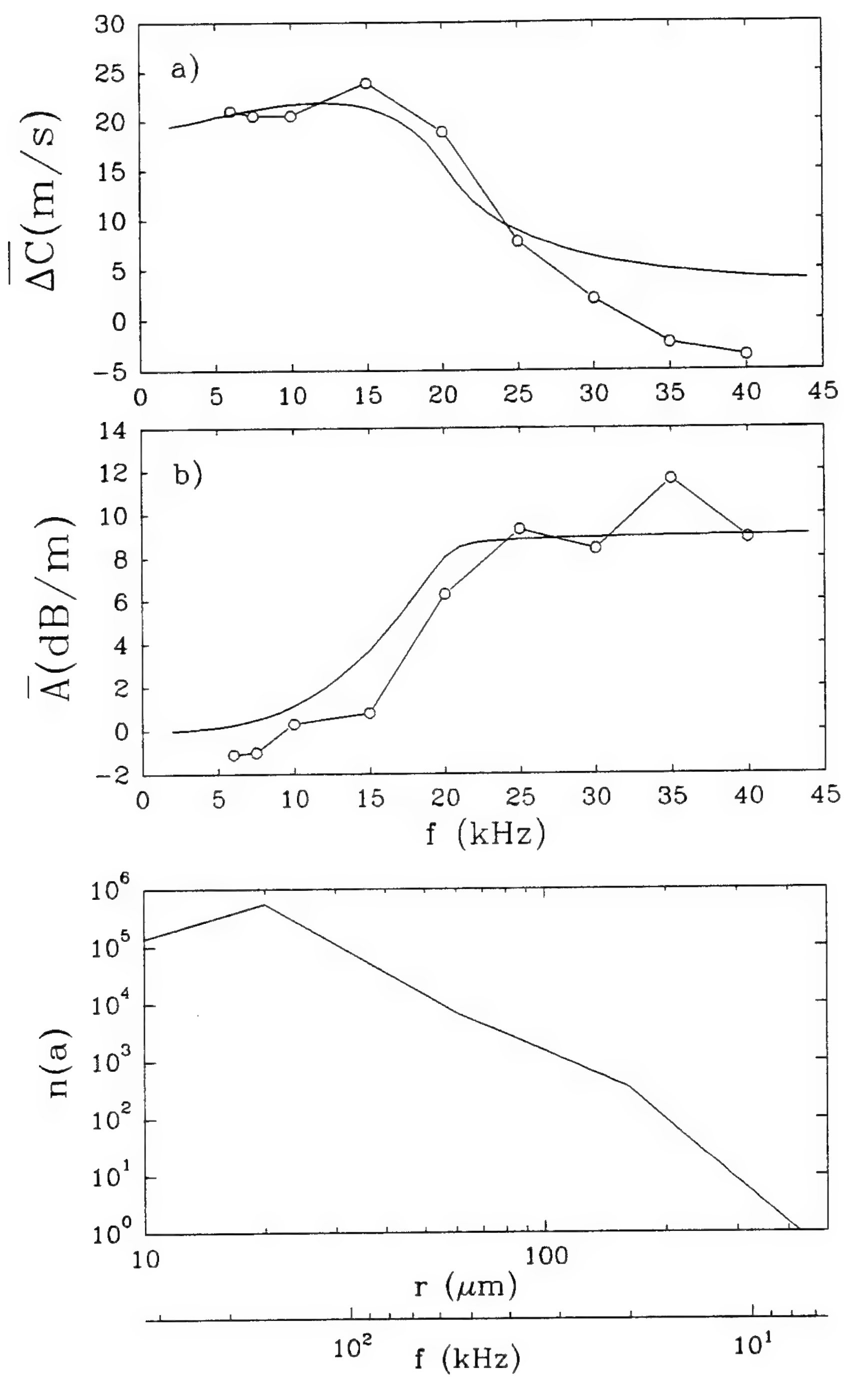


Figure 7.26: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency (from figure 7.19). The solid line in both plots is obtained by integrating equation B.18 and B.19 with a bubble population described by the parameters in table 7.4 and shown in c). Note that the horizontal axes in c) show both the bubble radius and the corresponding bubble resonance frequency. The number density of bubbles per unit micron increment is n(a). The solution was optimized to minimize the variance between the model and the data according to equation 7.6.

Figure 7.25 shows the sensitivity of the optimal solution given in table 7.4 to small changes in four of the bubble population parameters. For this figure, the parameters given in table 7.4 are used as the basic bubble population and one of the parameter is varied for each of the plots in figure 7.25. The standard deviation σ of the error between the measured and the modeled data is computed according to equation 7.6. The optimal solution is found when σ is minimized. In general, the standard deviations σ shown in figure 7.25 converge rapidly to a minimum except in the case of r_t. This is probably due to the fact that $r_t = 60\mu m$ (54kHz) will affect mostly frequencies outside the present range of measurements (6 to 40kHz) although some off-resonance contributions are expected (see below). The technique shows good sensitivity to small changes in the parameters which should be helpful, in the future, in accurately determining the character of the bubble population.

In figure 7.26 we compare the average measurements of sound-speed anomalies and attenuation (from figure 7.19) with the predictions given by equation B.18 and B.19 found in appendix B and the bubble population characterized by the parameters given in table 7.4. The fact that the sound-speed anomaly and the attenuation both agree relatively well with the data provides a reassuring check. The fit is not as good for the high-frequency sound-speed anomaly suggesting that the bubble population could have a more complex structure than the one used in the present model of figure 7.24.

Each feature of the bubble population controls some region of the complete sound-speed anomaly and attenuation curves shown in figure 7.26. In order to isolate the effect of each parameter, families of sound-speed anomaly and attenuation curves were generated and for each family only one parameter was varied. The most notable feature of figure 7.26a is the rapid transition from high to low sound-speed anomalies at approximately 20kHz. This transition is caused by the change in the slope of the bubble population at $r_c = 160\mu m$ (20kHz resonance). Figure 7.27a shows how the position of the transition is varied by varying r_c. The shape of the 'plateau' below 20kHz is controlled by the slope s_4 of the bubble population for bubbles of size greater than r_c. Figure 7.27b shows how the shape of the plateau varies with the slope s_4. Notice the small hump generated at 15kHz when the slope steepens. The rapid fall off at the transition (20kHz) is mostly controlled by s_3 the slope of the bubble population between r_t and r_c. Figure 7.27c shows how milder slopes generate a more rapid fall off at the transition. Finally, the position of r_t will also affect the shape of the sound-speed anomaly and attenuation curves even though the resonance frequency for a bubble of size $r_t = 60\mu m$ is 54kHz. This is due to off-resonance contributions and the effect of varying

r_t is shown to be felt down to approximately 25kHz (figure 7.27d). Both r_t and s_3 control the low values of sound-speed anomaly between 25 and 40kHz.

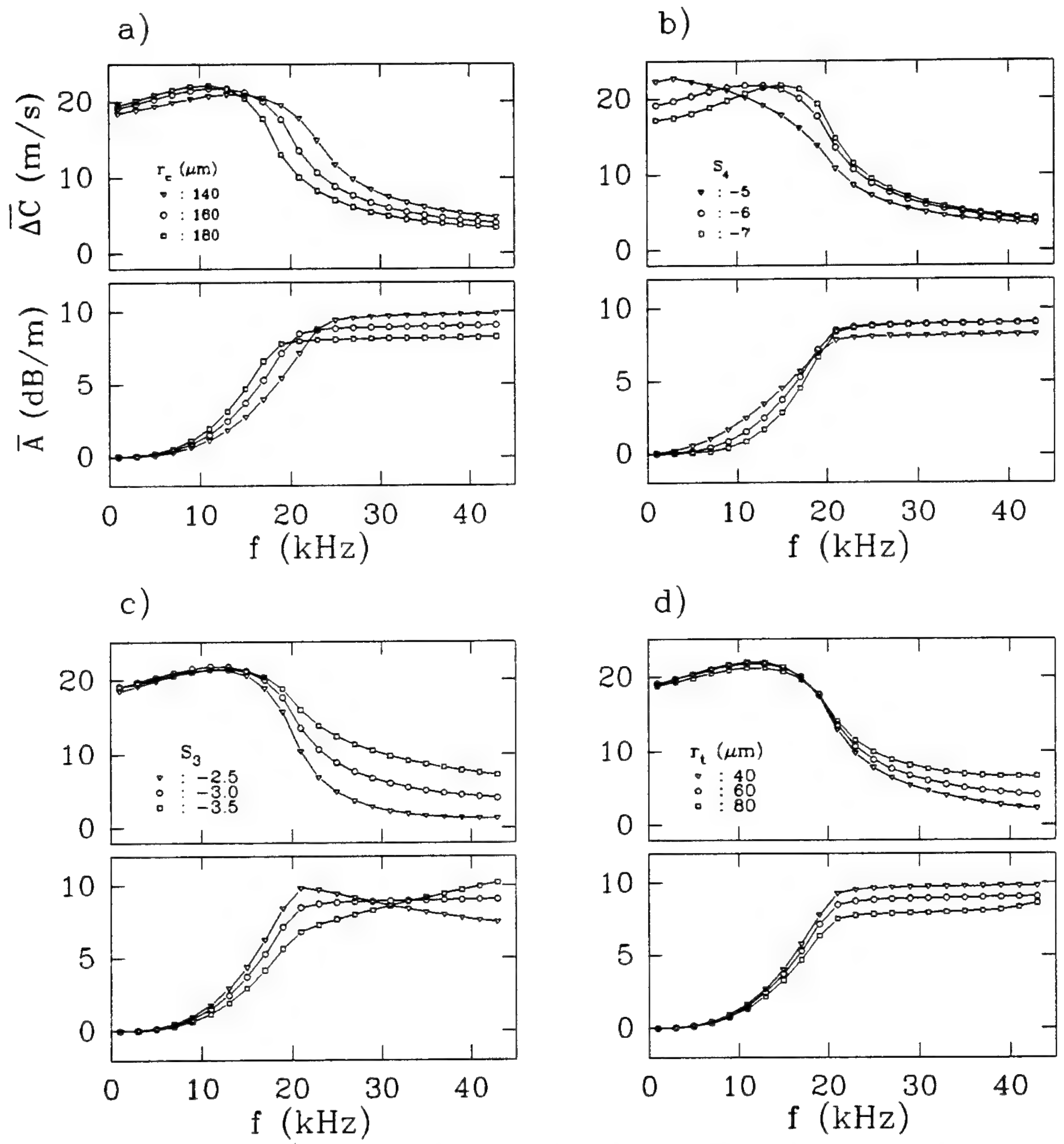


Figure 7.27: Sound-speed anomaly $\overline{\Delta c}$ (above) and attenuation $\overline{A}$ (below) as a function of frequency for different variations in the bubble population parameter a) r_c, b) s_4, c) s_3 and d) r_t. The starting bubble population is the one given in table 7.4 and figure 7.26c. Only one of the bubble population parameter is varied in each of the various graphs above. The symbol (o) corresponds to the original value and the symbols (∇) and (□) are the variations above and below the original value.

Perhaps the most important finding obtained from figure 7.26a is the presence of a transition to a relatively steep slope (a^{-6}) at a cutoff radius r_c=160µm. The steeper slope at large bubble radii is an essential requirement which ensures that the total amount of air in the water remains finite (see §1.4.2 for a detail discussion on this issue). The position of the cutoff is expected to change with sea-states and such behavior has been observed in bubble populations measured in a laboratory channel where the cutoff radius was pushed toward the larger bubble radii at higher wind speeds [Baldy, 1988].

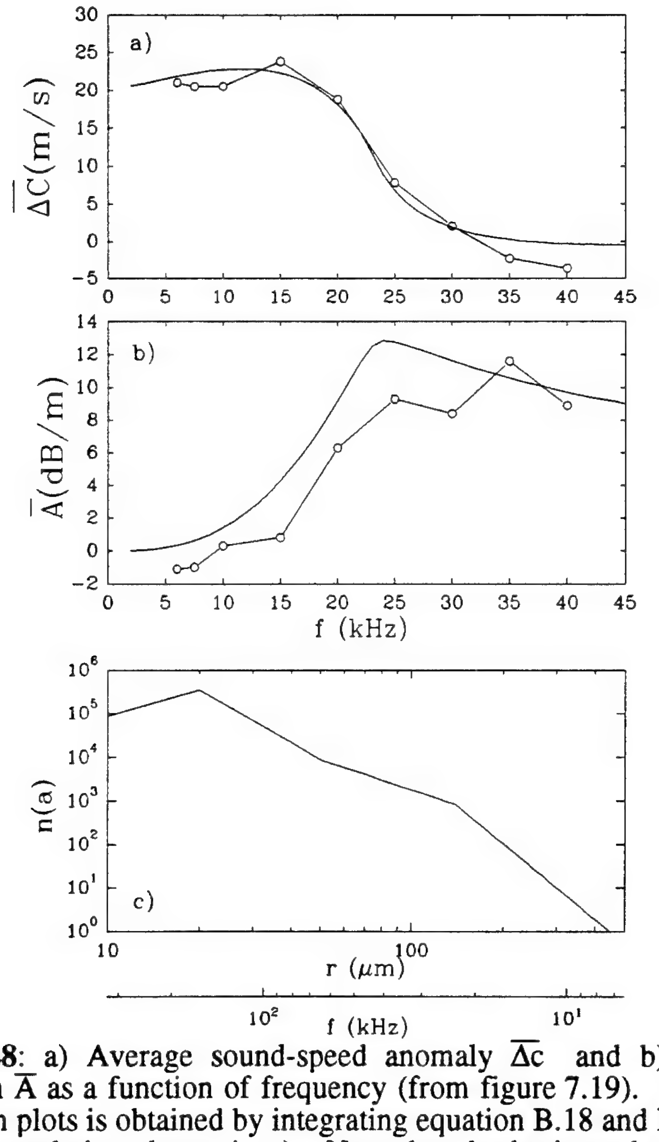


Figure 7.28: a) Average sound-speed anomaly $\overline{\Delta c}$ and b) average attenuation $\overline{A}$ as a function of frequency (from figure 7.19). The solid line in both plots is obtained by integrating equation B.18 and B.19 with a bubble population shown in c). Note that the horizontal axes in c) show both the bubble radius and the corresponding bubble resonance frequency. The number density of bubbles per unit micron increment is n(a). The solution was optimized to best fit only the sound-speed anomaly. The bubble population parameters are s_1=2, s_2=-4, s_3=-2.5, s_4=-6, r_p=20µm, r_t=50µm and r_c=140µm.

It is possible to improve on the fit of the sound-speed anomaly data shown in figure 7.26a (especially at high-frequencies) by selecting an optimization scheme based solely on the variance of the model with the sound-speed anomaly and disregarding the optimization with with respect to the attenuation data. The results of are shown in figure 7.28. The fit to the sound-speed anomaly data is better than the one shown in figure 7.26a but as expected the fit to the attenuation data is not as good but still remains adequate. There are still problems in fitting the highest frequency data (35 and 40kHz). This suggest that the bubble population may be more complex than has been assumed in the model of figure 7.24.

The measurements of sound-speed and attenuation at several frequencies using the composite pulse technique described in §7.7 permit one to infer the ambient bubble population in an effective manner. A single pulse is enough to get the necessary measurements to compute the bubble population but many repetitions will be required to obtain a statistically significant average. Indeed, previous time-series of the sound-speed anomaly and attenuation have shown that these signals vary significantly over periods of minutes indicating the necessity for long time-averages (O(1hr)). This may also explain the great variability of results in earlier bubble population studies which have typically used O(10min) averages.

7.9 Discussion and Summary

In this chapter, we have reported on the results from two field experiments on the measurement of sound-speed and sound attenuation near the ocean surface. Perhaps the most significant novelty of the present measurement technique is the ability to make simultaneous measurements at several different depths, starting as close as 0.5m, at frequencies down to 5kHz (60cm wavelength) and at a sample rate of 4Hz/ch. Furthermore, the technique for measuring the speed of sound is a 'direct' one and thus it avoids the many difficulties involved with inferring the sound-speed from bubble population measurements (see §1.4.2 and §1.8.1 for a discussion of these difficulties). Although the present measurements are limited to moderate wind conditions (8m/s or less), they have nevertheless furnished surprising results on the characteristic of the sound-speed and attenuation signals close to the ocean surface.

First and foremost, the time-series of the sound-speed anomaly and attenuation showed dramatic fluctuations over time periods on the order of minutes or less. The large fluctuations were attributed to bubble plumes/clouds drifting by the sensors. In two particular instances, a newly created bubble plume generated a sound-speed anomaly up to 800m/s ($\alpha=1.6\times10^{-4}$) at 0.5m; and a bubble cloud 25sec old generated an anomaly up to 400m/s. At a wind speed of 8m/s, several occurrences (O(10)) of sound-speed anomalies above 100m/s have typically been found in 20min time-series at a depth of 0.5m. These are the first reported time-series of sound-speed anomalies in the shallow ocean surface layer and the fluctuations measured are two to three orders of magnitude greater than previously reported time-averaged (10 to 20min) measurements [Medwin, 1974; Medwin et al., 1975a; Farmer and Vagle, 1989]. Few measurements of sound-speed anomalies have been observed at a depth of 1.5m and none at 2.0m for 8m/s wind conditions.

The signal at various depths was found to be highly correlated and mostly coherent at frequencies below 0.05Hz which corresponded to the frequency region where the signal was most energetic. The time-averaged sound-speed anomaly was found to decrease exponentially with depth with an e-folding depth of 0.18m and an extrapolated anomaly at the surface of 370m/s. These profile characteristics were obtained in wind conditions of 8m/s and a significant wave height of approximately 0.5m. In contrast, Farmer and Vagle [1989] have measured an exponential profile with an e-folding depth of 1.4m and an anomaly at the surface of 3m/s in wind conditions of 10m/s (and unreported significant wave height). The present measurements indicate that the sound-speed profile may be significantly more pronounced and shallower than the one previously reported by Farmer and Vagle [1989]. There are approximately one to two orders of magnitude differences on the e-folding depth and the surface anomaly between these two sound-speed profiles which were obtained in similar wind conditions. While some of the differences between these two sound-speed profiles are probably due to variations in the wave conditions, most of the differences may be due to problems associated with inferring sound-speed anomalies from bubble population measurements. Another possibility may be that the measurements of Farmer and Vagle [1989], which are performed with a sonar located 20 to 30m away from the surface, were not exactly at the depth reported. Small differences in measurement depth on the order of 10 to 50cm can make a dramatic difference in the magnitude of the signal measured.

The sound-speed anomaly was found to agree relatively well with an exponential probability distribution where the probability of occurrence of large anomalies decreased exponentially with the magnitude of the anomaly. As anticipated, the exponential decrease was found to be significantly higher with depth since the probability of finding a large sound-speed anomaly at greater depth is smaller. The exponential decrease of the probability distribution was also found to be higher at low sea-state and thus large sound-speed anomalies are less likely to be found in calmer sea conditions as one would expect.

The average sound-speed anomaly at a depth of 0.5m was found to increase with wind speed. The average anomalies were found to start increasing at wind speeds around 2 to 3m/s which is consistent with the wind-speed threshold for the onset of wave breaking in the ocean. At a wind-speed of 8m/s, the average sound-speed anomaly was approximately 20 to 25m/s and the extrapolation of the correlation up to 15m/s suggested that the average anomaly at a depth of 0.5m would be around 40 to 50m/s for these often encountered wind conditions.

The measurement technique presented in detail in §6 was expanded to include simultaneous measurements of the sound-speed and attenuation at multiple frequencies ranging from 6 to 40kHz. This novel technique makes use of a broad-band pulse which was band-pass filtered after completion of the experiment to extract the sound-speed and attenuation at the various frequencies. The technique could easily be improved to include real-time band-pass filtering and a much higher upper frequency range (up to 100kHz).

The results obtained showed that the sound-speed anomaly became non-dispersive (i.e. independent of frequency) below approximately 20kHz. Above 20kHz, the signal becomes highly dispersive and rapid changes in the sound-speed anomalies have been observed between 20 and 30kHz. At higher frequencies (35 and 40kHz), the average sound-speed anomaly became negative indicating that the propagation velocity was increasing from its bubble-free value. The attenuation measurements at high frequencies (20kHz and above) were found to follow closely the sound-speed anomalies at low frequencies (below 20kHz) at all measurement depths. It is the passage of bubble clouds which generate both the sound-speed anomaly and the attenuation signals and thus it comes as no surprise that both signals are similar.

The sound-speed and attenuation measurements were used to infer the bubble population characteristics. In particular, the shape of the bubble population at large bubble radii (say above of 100μm) was investigated. Previous studies of bubble population at sea had reported a slope of $a^{-2.5}$ to $a^{-3.0}$ for the larger bubble sizes. If this slope is correct, it necessarily requires an upper-cutoff at some large bubble radius where the slope must become steeper than a^{-3} in order to keep the amount of air in the water finite. In order to match the sound-speed and attenuation measurements at a wind speed of 8m/s, it was shown that the bubble population needed to have an upper-cutoff at

160μm and a slope of a^{-6} for bubbles beyond that radius. It was also pointed out that the location of the break point would probably be a function of the sea-state.

Thus, the measurement technique developed in this work has the potential to not only give accurate and rapid measurements of the sound-speed and attenuation near the ocean surface at low and high frequencies but it may also be capable, with some improvements in the processing technique, to yield bubble population measurements as well.

Chapter 8
Conclusions

Breaking waves charge the surface layer of the ocean with small air bubbles which, as argued in §1, are important for the enhancement of gas transfer across the air-sea interface and for acoustic propagation near the ocean surface. Although it is clear that breaking is the dominant mechanism by which air is entrained in the surface layer, it is not so easy to quantify the air entrainment itself and its effect on the upper-ocean. For example, basic questions such as how much air breaking waves can actually entrain and what the spatial and temporal distribution of the underwater air volume is are at the heart of the present work.

This thesis has reported on a series of laboratory and field measurements of air entrainment by breaking waves. The volumetric fractions of air found below a breaking wave can cover eight orders of magnitude from 10^{-8} to 10^{0} depending on the position below the breaking wave and the time elapsed since breaking occurred. At first, the air entrainment process evolves very rapidly after the onset of breaking. The bubble plumes formed by breaking waves typically contain high void-fractions (O(1%) and greater) during that initial period after breaking. At later times, the plume of bubbles continues to degas and to mix with the ambient fluid and the void-fraction continues to decrease. This thesis has reported on the development of instrumentation, and on laboratory and field measurements which have addressed the full range of void-fractions encountered in bubble plumes generated by breaking waves.

The first section of the thesis treated the measurement of high void-fractions. In Chapter 2, we developed novel instrumentation for the measurement of high void-fractions (0.3% and greater) in bubble plumes generated by breaking waves both in the laboratory and in the ocean. The instrument is based on the measurement of the electrical impedance of the bubbly mixture. The response time (100Hz), measuring volume (~400cm^3) and dynamic range (0.3% to 100%) of the instrument make it the only tool at present capable of mapping the high void-fraction distribution in bubble plumes during the initial period following onset of breaking. A particular characterisitic of the laboratory void-fraction probe is its ability to make unbias measurements of the void-fraction as close as 3cm to a free-surface.

In Chapter 3, a series of laboratory experiments were conducted in a 2D wave channel. The experiment consisted of mapping the evolution of the void-fraction field in bubble plumes generated by a range of breaking waves. Moments[1] of the void-fraction field were shown to scale well with the initially enclosed air volume at breaking and the energy dissipated by breaking. The results revealed that the bubble plume underwent rapid transformations within the first wave period after the onset of breaking. In particular, the bubble plume lost 95% of the total air entrained during that first wave period. This pointed out the importance of considering the transient nature of the air entrainment process in modeling the contribution of bubbles to air-sea gas transfer, especially for gases such as carbon dioxide which can equilibrate within a second or less.

It was shown that a significant fraction (30 to 50%) of the energy dissipated by breaking was used in entraining the bubbles against their buoyancy thus giving direct evidence of a strong dynamical coupling between the surface wave evolution and the air entrainment process. The energy dissipated by breaking was also found to correlate with the total volume of air entrained. Since dissipation also scales with the acoustic energy radiated by breaking [Loewen and Melville, 1991a] this suggested the possibility of monitoring and quantifying air entrainment at the ocean surface with passive acoustic remote sensing techniques.

The mean void-fraction in the bubble plume was found to remain above 1% within the first wave period after breaking. The measurements therefore showed that the bubble plume formed a compact region of much lower compressibility than the surrounding fluid. Previous investigators had proposed that if such bubble assemblage existed they could generate low-frequency sound by undergoing harmonic volume oscillations at frequencies much lower than the constituent bubbles. This hypothesis was confirmed by comparing the predicted resonance frequencies of the plume obtained from the void-fraction measurements (§3 and §4) with independent measuremrents of the low-frequency sound generated by the laboratory breaking waves [Loewen and Melville, 1993]. Thus, the results showed that volume oscillations of bubble plumes generated by breaking waves are a source of low-frequency sound. Whether this source dominates the oceanic ambient sound at low-frequencies remains to be shown.

In Chapter 4, we essentially repeated the experiment of §3 in a large 3D wave basin. The various moments of the void-fraction field were found to be comparable to those of

[1] Moments of the void-fraction field include the volume of air entrained, the cross-sectional area of the plume, the mean void-fraction throughout the plume and the potential energy of the plume.

the 2D experiment (§3) up to half a wave period after breaking. The differences at later time (beyond 0.5T) were attibuted to 3D effects. The kinematics of the plume were investigated and it was found that the centroid of the plume moved horizontally at approximately 0.7C where C is the phase speed of the wave. The vertical position of the centroid was found to deepen at 0.2H/T with respect to the free surface where H and T are the wave height at breaking and the wave period respectively.

The radial distribution of the void-fraction within the bubble plume was found to be described by simple functions which depended on the bubble plume radius and time. These results, coupled with recent models of bubble resonance which include the radial variations in the void-fraction [Koller and Shankar, 1992], should help in determining whether these variations are likely to yield resonance frequencies substantially different from those predicted using a uniform void-fraction.

In Chapter 5, we presented results on the measurement of void-fraction near the ocean surface. Measurements at a depth of 20 and 50cm showed occasional high void-fractions up to 24% which were caused by waves breaking immediately above the sensors. These results were several orders of magnitude greater than time-averaged values previously reported [Walsh and Mulheran, 1987] and they showed that the void-fraction field was highly variable in time and space. Void-fraction events above the noise threshold of the instrument were found to be sporadic at a depth of 20cm and no occurences were detected at 80cm for wind speeds up to 15m/s. This pointed out the importance of developing a more sensitive instrument to measure smaller void-fractions. The void-fraction probe was also shown to be a useful instrument for the study of breaking wave statistics. The fraction of breaking waves detected with the void-fraction probe was found to increase with wind speed and the results were consistent with the lower-bound data of previous studies conducted with different techniques. Video photography near the surface revealed the presence of surprisingly large bubbles (at least 6mm in radius). These large bubbles have not yet been quantified and the role they play in gas transfer and upper ocean acoustics has not been addressed.

The second section of the thesis dealt with the measurement of very low void-fractions (10^{-7} and above). In Chapter 6 we developed an instrument to measure the low-frequency sound-speed (and thus the void-fraction). The measurement technique is based on the propagation velocity of low-frequency acoustic pulses. The measurement system was thoroughly tested in the laboratory and in a lake and the resolution was determined to be better than ±2m/s. Perhaps the most significant novelty of this measurement technique is the ability to make simultaneous real-time measurements at several depths, starting as close as 0.5m, at frequencies down to 5kHz and at a sample rate of 4Hz/ch. The technique provides a direct measurement of the sound-speed and thus it avoids the many difficulties involved with inferrring the sound-speed from bubble population measurements.

In Chapter 7 we reported on the results of two field experiments on the measurement of sound-speed and sound attenuation near the ocean surface. The time-series of the sound-speed anomaly and attenuation showed dramatic fluctuations over time periods on the order of minutes or less. The large fluctuations were attributed to bubble plumes/clouds drifting by the sensors and in one instance when a wave broke immediately above the instrument, the anomaly reached 800m/s (1.6×10^{-4} void-fraction). The fluctuations dominated the long-term averages and they were found to be two to three order of magnitude greater than previously reported time-averaged measurements [Medwin, 1974; Medwin et al., 1975; Farmer and Vagle, 1989]. Since the fluctuations are the dominant part of the sound-speed anomaly and attenuation signal, it was concluded that characterization of these signals in terms of their long-term averages may not always be appropriate.

The signal at various depths was found to be highly correlated and mostly coherent at frequencies below 0.05Hz which corresponded to the frequency region where the signal was most energetic. The time-averaged (20min) sound-speed anomaly, for wind conditions of 8m/s and SWH of 0.5m, was found to decrease with depth with an e-folding depth of 0.18m and an extrapolated anomaly at the surface of 370m/s. This sound-speed profile is significantly more pronounced and shallower than the one previously reported by Farmer and Vagle [1989] in 10m/s winds (and unreported significant wave height).

The average sound-speed anomaly at 0.5m was found to increase with wind speed and the correlation suggested averaged anomalies as high as 40 to 50m/s at wind speeds of 15m/s. The probability distribution of the sound-speed anomaly signal was found to agree relatively well with an exponential distribution. The exponential decrease of the probability distribution was higher with an increase in depth and a decrease in wind speed. Thus, the probability of finding a sound-speed anomaly of a certain magnitude was lower with an increase in depth and a decrease in wind speed.

The measurement technique of §6 was expanded to include simultaneous measurements of the sound-speed and attenuation at multiple frequencies using a broad-band acoustic pulse. The results showed that the sound-speed was non-dispersive below

20kHz for wind conditions up to 8m/s. The attenuation at high frequencies (20kHz and above) was found to closely follow the sound-speed anomalies at low-frequencies (below 20kHz) and this was attributed to the passage of bubble clouds which generate both the attenuation and anomaly signals at the same time. The sound-speed and attenuation measurements were used to infer the bubble population characterisitics for large bubble radii (say above 100μm). The results showed that the population of large bubbles has an upper cutoff at which point the slope of the bubble population increases significantly. For the 8m/s wind conditions observed during the experiments, we found this cutoff to be at a radius of 160μm and the change in slope at the transition went from a^{-3} to a^{-6}. The presence of this transition to a steeper slope at large bubble radii as not been documented in previous field studies of bubble population even though its existence is required in order to keep the amount of air in the water finite.

References

Agrawal, Y.C., Terray, E.A., Donelan, M.A., Hwang, P.A., Williams III, A.J., Drennan, W.M., Kahma, K.K. & Kitaigorodski, S.A. 1992 Enhanced dissipation of kinetic energy beneath surface waves. *Nature* **359**, 219-220.

Baldy, S. 1988 Bubbles in the close vicinity of breaking waves: statistical characteristics of the generation and dispersion mechanism. *J. Geophys. Res.* **93**, 8239-8248.

Banner, M.L. & Cato, E.H. 1988 Physical mechanisms of noise generation by breaking waves - laboratory study. In *Sea surface sound*, edited by Kerman, B.R., Kluwer Academic, Dordrecht, 237-246.

Banner, M.L. & Peregrine, D.H. 1993 Wave breaking in deep water. *Annu. Rev. Fluid Mech.* **15**, 149-178.

Bell, T.H. 1989 Bubble clouds produced by breaking waves. Technical report No.2877, ORI, Rockville, Maryland.

Blanchard, D.C & Woodcock, A.H. 1957 Bubble formation and modification in the sea and its meteorological significance. *Tellus* **9**, 145-158.

Blanchard, D.C. & Syzdek, L.D. 1972 Concentrations of bacteria in jet drops from bursting bubbles. *J. Geophys. Res.* **77**, 5087-5099.

Blanchard, D.C. 1983 The production, distribution and bacterial enrichment of the sea-salt aerosol. In *Air-Sea Exchange of Gases and Particles*, edited by Liss, P.S., & W. George N. Slinn, Reidel, Dordrecht, 407-454.

Broecker, H.Ch. & Siems, W. 1984 The role of bubbles for gas transfer from water to air at higher windspeeds. Experiments in the wind-wave facility in Hamburg. In *Gas transfer at water surfaces*, edited by Brutsaert, W. & Jirka, G.H., Reidel, Dordrecht, 229-236.

Broecker, W.S. & Peng, T.H. 1974 Gas exchange rates between air and sea. *Tellus* **21**, 16-35.

Broecker, W.S. & Peng, T.H. 1984 Gas exchange measurements in natural systems. In *Gas transfer at water surfaces*, edited by Brutsaert, W. & Jirka, G.H., Reidel, Dordrecht, 479-494.

Broecker, W.S., Ledwell, J.R., Takahashi, T., Weiss, R., Merlivat, L., Memery, L., Peng, T-H, Jahne, B. & Munnich, K.O. 1886 Isotopic versus micrometeorologic ocean CO_2 fluxes: a serious conflict. *J. Geophys. Res.* **91**, 10517-10527.

Buckingham, M.J. 1991 On acoustic transmission in ocean-surface waveguides. *Phil. Trans. R. Soc. Lond.* **A335**, 513-555.

Carey, W.M. & Browning, D. 1988 Low frequency ocean ambient noise: measurements and theory. In *Sea surface sound*, edited by Kerman, B.R., Kluwer Publishers, Dordrecht, 361-376.

Carey, W.M. & Fitzgerald, J.W. 1993 Low frequency noise from breaking waves. In *Natural physial sources of underwater sound*, edited by Kerman, B.R., Kluwer Publishers, Dordrecht, 277-304.

Carey, W.M., Fitzgerald, J.W., Monahan, E.C. & Wang, Q. 1993 Measurement of the sound produced by a tipping trough with fresh and salt water. *J. Acoust. Soc. Am.*, submitted.

Clay, C.S. & Medwin, H. 1977 *Acoustical Oceanography*, John Wiley & Sons.

Commander, K.W. & McDonald, R.J. 1991 Finite-element solution of the inverse problem in bubble swarm acoustics. *J. Acoust. Soc. Am.* **89**, 592-597.

Commander, K.W. & Prosperetti, A. 1989 Linear pressure waves in bubbly liquids: Comparison between theory and experiments. *J. Acoust. Soc. Am.* **85**, 732-746.

Crawford, G.B. & Farmer, D.M. 1987 On the spatial distribution of ocean bubbles. *J. Geophys. Res.* **92**, 8231-8243.

Crowther, P.A. 1988 Bubble noise creation mechanisms. In *Sea surface sound*, edited by Kerman, B.R., Kluwer Publishers, Dordrecht, 131-150.

d'Agostino, L. & Brennen, C.E. 1983 On the acoustical dynamics of bubble clouds. *ASME Cavitation and Multiphase Flow Forum*, 8-13.

d'Agostino, L. & Brennen, C.E. 1988 Acoustical absorbtion and scattering cross section of spherical bubble clouds. *J. Acoust. Soc. Am.* **84**, 2126-2134.

Dalen, J. & Lovick, A. 1981 The influence of wind induced bubbles in water. *J. Acoust. Soc. Am.* **69**, 1653-1659.

Dean, R.G. & Dalrymple, R.A. 1984 *Water Wave Mechanics for Engineers and Scientists.* Prentice-Hall Englewood Cliffs, NJ.

Del Grosso, V.A. 1974 New equation for the speed of sound in natural waters (with comparisons to other equations). *J. Acoust. Soc. Am.* **56**, 1084-1091.

Ding, L., 1992 Acoustical studies of breaking waves in the open ocean. Ph.D. thesis, University of Victoria & Institute of Ocean Science, B.C. Canada.

Etcheto, J. & Merlivat, L. 1988 Satellite determination of the carbon dioxide exchange coefficient at the ocean-atmosphere interface: a first step. *J. Geophys. Res.* **93**, 15669-15678.

Farmer, D.M. & Lemon, D.D. 1984 The influence of bubbles on ambient noise in the ocean at high wind speeds. *J. Phys. Oceanog.* **14**, 1762-1778.

Farmer, D.M. & Vagle, S. 1988 On the determination of breaking surface wave distributions using ambient sound. *J. Geophys. Res.* **93**, 3591-3600.

Farmer, D.M. & Vagle, S. 1989 Waveguide propagation of ambient sound in the ocean surface bubble layer. *J. Acoust. Soc. Am.* **86**, 1897-1908.

Farmer, D.M. & Ding, L., 1992 Coherent acoustical radiation from breaking waves, *J. Acoust. Soc. Am.* **93**, 397-402.

Farmer, D.M., McNeil, C.L. & Johnson, B.D. 1993 Evidence for the importance of bubbles to the enhancement of air-sea flux. *Nature*, submitted.

Gill, A.E. 1982 *Atmosphere-Ocean Dynamics.* Academic Press, San-Diego, CA, 662pp.

Glotov, V.P., Kolobayev, P.A. & Neuimin, G.G. 1962 Investigation of scattering of sound by bubbles generated by an artificial wind in sea water and the statistical distribution on bubbles sizes. *Sov. Phys. Acoust.*, Engl. Transl., **7**, 341-345.

Henyey, F.S. 1991 Acoustic scattering from microbubble plumes in the 100Hz to 2kHz region. *J. Acoust. Soc. Am.* **90**, 399-405.

Herwig, H. & Nutzel, B. 1989 The influence of bubbles on acoustic propagation and scattering. In *Underwater Acoustic Data Processing*, edited by Chan, Y.T., Kluwer Publishers, Dordrecht, 105-111.

Hill, K.D. & Wood, D.J. 1988 The dynamic response of the two-electrode conductivity cell. *J. Ocean. Eng.* **13**, 118-123.

Holthuijsen, L.H. & Herbers, H.C. 1986 Statistics of breaking waves observed as whitecaps in the open ocean. *J. Phys. Oceanog.* **16**, 290-297.

Hollett, R.D. 1993 Observations of underwater sound at frequencies below 1500Hz from breaking waves. *J. Acoust. Soc. Am.*, submitted.

Horowitz, P. & Hill, W. 1989 *The Art of Electronics*, Cambridge University Press, Cambridge, England.

Huang, W.H. 1990 Wave tank automation. Internal report, Department of Civil & Environmental Engineering, Massachusetts Institute of Technology, MA.

Hwang, P.A., Hsu, Y.-H., & Wu, J. 1990 Air bubbles produced by breaking wind waves: a laboratory study. *J. Phys. Oceanog.* **20**, 19-28.

Jahne, B, Wais, T. and Barabas, M. 1984 A new optical bubble measuring device; a simple model for bubble contribution to gas exchange. In *Gas transfer at water surface.* edited by Brutsaert, W. and Jrika, G.H., Reidel Publishers, Dordrecht, 237-246.

Jahne, B., Wais, T., Memery, L., Caulliez, G., Merlivat, L., Munnich, K.O. and Coantic, M. 1985 He and Rn gas echange experiments in the large wind-wave facility of IMST. *J. Geophys. Res.* **90**, 11989-11997.

Jahne, B., Munnich, K.O., Rainer B., Dutzi, A., Huber, W. & Libner, P. 1987 On the parameters influencing air-water gas exchange. *J. Geophys. Res.* **92**, 1937-1949.

Jessup, A.T., Melville, W.K. & Keller, W.C. 1991a Breaking waves affecting microwave backscatter: 1. Detection and verification. *J. Geophys. Res.* **96**, 20547-20559.

Jessup, A.T., Melville, W.K. & Keller, W.C. 1991b Breaking waves affecting microwave backscatter: 2. Dependence on wind and wave conditions. *J. Geophys. Res.* **96**, 20561-20569.

Johnson, B.D. & Cooke, R.C. 1979 Bubble populations and spectra in coastal waters: a photographic approach. *J. Geophys. Res.* **84**, 3761-3766.

Kalvoda, P.M. 1992 Macrobubble clouds produced by breaking wind waves. M.Sc. thesis, University of Delaware, Delaware.

Karplus, H.B. 1958 The velocity of sound in a liquid containing gas bubbles. United States Atomic Energy Commission. (Available from NTIS document number C00-248).

Katsaros, K.B. & Ataktürk, S.S. 1991 Dependence of wave breaking statistics on wind stress and wave development. Presented at the *Breaking Waves Sympium*, International Union of Theoretical and Applied Mechanics, Sydney, Australia.

Kerman, B.R. 1984 Underwater sound generation by breaking waves. *J. Acoust. Soc Am.* **75**, 149-165.

Kerman, B.R. (Ed) 1988 *Sea surface sound.* Kluwer Publishers, Dordrecht, 639pp.

Kerman, B.R. (Ed) 1993 *Natural physical sources of underwater sound.* Kluwer Publishers, Dordrecht, 750pp.

Khoo, B.C. & Sonin, A.A. 1992 Augmented gas exchange across wind-sheared and shear-free air-water interfaces. *J. Geophys. Res.* **97**, 14 413-14 415.

Knudsen, V.O., Alford, R.S. & Emling, J.W. 1948 Underwater ambient noise. *J. Marine Res.* **7**, 410-429.

Kolaini, A., Roy, R.A. & Crum, L.A. 1991 An investigation of the acoustic emissions from a bubble plume. *J. Acoust. Soc. Am.* **89**, 2452-2455.

Koller, D., Li, Y., Shankar, P.M. & Newhouse, V.L. 1992a High-speed bubble sizing using the double frequency technique for oceanographic applications. *J. Ocean. Eng.* **17**, 288-291.

Koller, D.P. & Shankar, P.M. 1992b Low-frequency oscillations of bubble plumes. *J. Acoust. Soc. Am.* **S92**, 2349.

Koller, D.P. & Shankar, P.M. 1993 Low-frequency oscillations of bubble plumes. *J. Acoust. Soc. Am.* **93**, 1362-1364.

Kolovayev, P.A. 1976 Investigation of the concentration and statistical size distribution of wind-produced bubbles in the near-surface ocean. *Oceanology*, Engl. Transl., **15**, 659-661.

Lamarre, E. & Melville, W.K. 1991 Air entrainment and dissipation in breaking waves. *Nature* **351**, 469-472.

Lamarre, E. & Melville, W.K. 1992 Instrumentation for the measurement of void-fraction in breaking waves: laboratory and field results. *J. Ocean. Eng.* **17**, 204-205.

Lamarre, E. & Melville, W.K. 1993 Void-fraction measurements and sound speed fields in bubble plumes generated by breaking waves. *J. Acoust. Soc. Am.* submitted.

Large, W.G. & Pond, S. 1981 Open ocean momentum flux measurements in moderate to strong winds. *J. Phys. Oceanogr.* **11**, 324-336.

Laville, F., Abbott, G.D. & Miller, M.J. 1991 Underwater sound generation by rainfall. *J. Acoust. Soc.* **89**, 715-721.

Liss, P.S. & Merlivat, L. 1986 Air-sea gas exchange: introduction and synthesis. In *The role of air-sea exchange in geochemical cycling*, edited by P. Buat-Menard, Reidel Publishers, Dordrecht, pp. 113-127.

Longuet-Higgins, M.S. & Turner, J.S. 1974 An 'entraining plume' model of a spilling breaker. *J. Fluid Mech.* **63**, 1-20.

Longuet-Higgins, M.S. & Smith, N.D. 1983 Measurement of breaking waves by a surface jump meter. *J. Geophys. Res.* **88**, 9823-9831.

Loewen, M.R. and Melville, W.K. 1991a Microwave backscatter and acoustic radiation from breaking waves. *J. Fluid Mech.* **224**, 601-623.

Loewen, M.R. and Melville, W.K. 1991b A model of the sound generated by breaking waves. *J. Acoust. Soc. Am.* **90**, 2075-2080.

Loewen, M.R. 1991 Laboratory measurements of the sound generated by breaking waves. Ph.D. thesis, Massachusetts Institute of Technology & Woods Hole Oceanographic Institution, MA.

Loewen, M.R. and Melville, W.K. 1993 An experimental investigation of the collective oscillations of bubble plumes entrained by breaking waves. *J. Acoust. Soc. Am.*, submitted.

Lu, N.Q., Prosperetti, A. & Yoon, S.W. 1990 Underwater noise emissions from bubble clouds, *J. Ocean. Eng.* **15**, 275-281.

McCammon, D.F. & McDaniel, S.T. 1990 Spectral spreading from surface bubble motion. *J. Ocean. Eng.* **15**, 95-100.

McDonald, B.E. 1991 Echoes from vertically striated subresonant bubble clouds: a model for ocean surface reverberation. *J. Acoust. Soc. Am.* **89**, 617-622.

Maxwell, J.C. 1892 *A Treatise on Electricity and Magnetism*, vol. 1, Clarendon Press, Oxford, England.

Medwin, H. 1970 In situ acoustic measurements of bubble populations in coastal waters. *J. Geophys. Res.* **75**, 599-611.

Medwin, H. 1974 Acoustic fluctuations due to microbubbles in the near-surface ocean. *J. Acoust. Soc. Am.* **75**, 1100-1104.

Medwin, H., Fitzgerald, J. & Rautman, G. 1975a Acoustic miniprobing for ocean microstructure and bubbles. *J. Geophys. Res.* **80**, 405-413.

Medwin, H. 1975b Speed of sound in water: a simple equation for realistic parameters. *J. Acoust. Soc. Am.* **58**, 1318-1319.

Medwin, H. 1977 In situ acoustic measurements of microbubbles at sea. *J. Geophys. Res.* **82**, 971-976.

Medwin, H. & Breitz, N.D. 1989 Ambient and transient bubble spectral densities in quiescent seas and under spilling breakers. *J. Geoph. Res.* **94**, 12751-12759.

Medwin, H. & Beaky, M.M. 1989 Bubble sources of the Knudsen sea noise spectra. *J. Acoust. Soc. Am.* **86**, 1124-1130.

Medwin, H. & Daniel Jr. A.C. 1990 Acoustical measurements of bubble production by spilling breakers. *J. Acoust. Soc. Am.* **88**, 408-412.

Melville, W.K. & Rapp, R.J. 1985 Momentum flux in breaking waves. *Nature* **317**, 514-516.

Melville, W.K., Loewen, M.R., Felizardo, F.C., Jessup, A.T. & Buckingham, M.J. 1988 Acoustic and microwave signatures of breaking waves. *Nature* **336**, 54-59.

Melville, W.K., Loewen, M.R. and Lamarre, E. 1992 In *Breaking waves,* edited by Banner, M.L. & Grimshaw, R.H.J, IUTAM Symposium in Sydney, Australia, Sringer-Verlag, Berlin.

Melville, W.K., Loewen, M.R. and Lamarre, E. 1993 Bubbles, noise and breaking waves: a review of laboratory experiments. In *Natural physical sources of underwater sound,* edited by Kerman, B.R., Kluwer academic publishers, 483-501.

Melville, W.K. 1992 The role of wave breaking in air-sea interaction. International Congress of Theoretical and Applied Mechanics, Haifa, Israel.

Memery, L. & Merlivat, L. 1985a Modelling of gas flux through bubbles at the air-water interface. *Tellus* **37B**, 272-285.

Memery, L. & Merlivat, L. 1985b Influence of gas transfer on the CO_2 uptake by the ocean. *J. Geophys. Res.* **90**, 7361-7366.

Merlivat, L. & Memery, L. 1983 Gas exchange across an air-water interface: experimental results and modeling of bubble contribution to transfer. *J. Geoph. Res.* **88**, 707-724.

Minneart, M. 1933 On musical air bubbles and the sound of running water. *Phil. Mag.* **16**, 235-248.

Monahan, E.C. & McNiocaill, G. (Eds) 1986 *Ocean Whitecaps and their role in air-sea exchange processes.* Reidel Publishers, Dordrecht, 294pp.

Monahan, E.C. & Wang, Q. 1992 Temporal evolution of the bubble plumes generated by breaking waves: acoustical implications. *J. Acoust. Soc. Am.* **S91**, S2452-2455.

Nystuen, J.A. 1986 Rainfall measurements using underwater ambient sound. *J. Acoust. Soc. Am.* **79**, 972-982.

O'Hearn, T.J., d'Agostino L. & Acosta, A.J. 1988 Comparison of holographic and Coulter counter measurements of cavitation nuclei in the ocean. *J. Fluids Eng.* **110**, 200-207.

Olsen, H.O. 1967 Theoretical and experimental investigation of impedance void meters. Thesis, Kjeller Research Establishment, University of Olso, Norway, 1967.

Omta, R. 1987 Oscillations of a cloud of bubbles of small and not so small amplitude. *J. Acoust. Soc. Am.* **82**, 1018-1024.

Prosperetti, A. 1988 Bubble dynamics in oceanic ambient noise. In *Sea surface sound*, edited by Kerman, B.R., Kluwer Publishers, Dordrecht, 151-171.

Pumphrey, H.C. & Ffowcs Williams, J.E. 1990 Bubbles as sources of ambient noise. *J. Oceanic Eng.* **15**, 268-274.

Quazi, A.H. 1981 An overview on the time delay estimate in active and passive systems for target localization. IEEE Trans. ASSP, **29**, 527-533.

Rapp, R.J. 1986 Laboratory measurement of deep water breaking waves. Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA.

Rapp, R.J. and Melville, W.K. 1990 Laboratory measurements of deep-water breaking waves. *Phil. Trans. R. Soc. Lond.* **A331**, 735-800.

Rino, C.L. & Ngo, H.D. 1991 Low-frequency acoustic scatter from subsurface bubble clouds. *J. Acoust. Soc. Am.* **90**, 406-415.

Roether, W. 1986 Field measurements of gas exchange. In *Dynamic processes in the chemistry of the upper ocean.* edited by Burton, J.D., Brewer, P.G. and Chesselet, R., Plenum, New-York, 117-128.

Roether, W. & Kromer, B. 1984 Optimum application of the radon deficit method to obtain air-sea gas exchange rates. In *Gas transfer at water surface.* edited by Brutsaert, W. and Jrika, G.H., Reidel Publishers, Dordrecht, 447-457.

Roy, R.A., Carey, W., Nicholas, M., Jeffrey, S. & Crum, L.A. 1992 Low-frequency scattering from submerged bubble clouds. *J. Acoust. Soc. Am.* **92**, 2993-2996.

Rubesch, J.E. 1990 The Design of a hybrid resistance-capacitance 50MHz impedance void-fraction meter. Master thesis, Mechanical Engineering, Massachusetts Institue of Technology, MA.

Ruggles, A.E. 1987 The propagation of pressure perturbationsin bubbly air/water flows. Ph.D. thesis, Mechanical Engineering, Rensselaer Polytechnic Institute, NY.

Silberman, E. 1957 Sound velocity and attenuation in bubbly mixtures measured in standing wave tubes. *J. Acoust. Soc. Am.* **29**, 925-933.

Smith, J.A. 1992 Observed growth of Langmuir circulation. *J. Geophys. Res.* **97**, 5651-5664.

Smith, S.D. & Jones, E.P. 1985 Evidence for wind-pumping of air-sea gas exchange based on direct measurements of CO_2 fluxes., *J. Geophys. Res.* **90**, 869-875.

Smith, S.D. & Jones, E.P. 1986 Isotopic and micrometeorological ocean CO_2 fluxes: different time and space scales., *J. Geophys. Res.* **91**, 10529-10532.

Su, M.-Y., Green, A.W. & Bergin, M.T. 1984 Experimental studies of surface breaking waves and air entrainment. In *Gas transfer at water surface.* edited by Brutsaert, W. and Jrika, G.H., Reidel Publishers, Dordrecht, 211-219.

Su, M.-Y., Ling, S.C. & Cartmill, J. 1988 Optical microbubble measurement in the North Sea. In *Sea surface sound*, edited by Kerman, B.R., Kluwer Publishers, Dordrecht.

Su, M.-Y., Burge, R. & Cartmill, J. 1993 Measurement of near-surface microbubble density during SWADE. *J. Geophys. Res.* submitted.

Su, M.-Y., Burge, R. & Cartmill, J. 1993 Breaking wave and void fraction statistics during SWADE. *J. Phys. Oceanog.* submitted.

Sundquist, E.T. 1993 The global carbon dioxide budget. *Science* **259**, 934-941.

Thorpe, S.A. & Humphries, P.N. 1980 Bubbles and breaking waves. Nature **283**, 463-465.

Thorpe, S.A. 1982 On the clouds of bubbles formed by breaking waves in deep water and their role in air-sea gas transfer. *Phil. Trans. Roy. Soc. Lond.* **A304**, 155-210.

Thorpe, S.A. & Hall, J.A. 1987 Bubble clouds and temperature anomalies in the upper oceans. *Nature* **328**, 48-51.

Thorpe, S.A. 1992 Bubble clouds and the dynamics of the upper-ocean. *Q. J. R. Meteorol. Soc.* **118**, 1-22.

Toba, Y.H., Kunishi, K., Nishi, S., Kawai, S., Shimada, Y. & Shibata, N., 1971 Study on air-sea boundary processes at the Shirahama oceanographic tower station. Disaster Prevention Institute, *Kyoto Univ. Annals* 148, 519-531.

Toba, Y., Masayuki, M., Okuda, K. & Kawai, S. 1975 Forced convection accompanying wind waves. *J. Oceanog. Soc. Japan* **31**, 192-198.

Updegraff, G.E. & Anderson, V.C. 1991 Bubble noise and wavelet spills recorded 1m below the ocean surface. *J. Acoust. Soc. Am.* **89**, 2264-2279.

Urick, R.J. 1975 *Principles of underwater sound for engineers.* McGraw-Hill, New-York.

Urick, R.J. 1986 *Ambient noise in the sea.* Peninsula Publishers, Los Altos, CA.

U.S. Army Corps of Engineers 1984 *Shore protection manual*, vol 1, Dept. of the Army, Washington DC.

Vagle, S. 1989 An acoustical study of the upper-ocean boundary layer. Ph.D. thesis, University of Victoria & Institute of Ocean Science, B.C. Canada.

Vagle, S., Large, W.G., Farmer, D.M. 1990 An evaluation of the WOTAN technique of inferring oceanic winds from underwater ambient sound. *J. Atm. and Oceanic Tech.* **7**, 576-595.

Vagle, S, & Farmer, D.M. 1992 The measurement of bubble-size distributions by acoustical backscatter. *J. Atmos. Oceanic Tech.* **9**, 630-644.

Wallace, D.W.R & Wirick, C.D. 1992 Large air-sea gas fluxes associated with breaking waves. *Nature* **356**, 694-696.

Wallace, G.T. & Duce, R.A. 1978 Open ocean transport of particulate trace metals. *Deep Sea Res.* **25**, 827-835.

Walsh, A.L. & Mulhearn, P.J. 1987 Photographic measurements of bubble populations from breaking wind waves at sea. *J. Geophys. Res.* **92**, 14553-14565.

Wanninkhof, R., Ledwell, J.R. & Broecker, W.S. 1985 Gas exchange-wind speed relation measured with sulfur hexafluoride on a lake. *Science* **227**, 1224-1226.

Watson, A.J., Upstill-Goddard, R.C., & P.S. Liss 1991 Air-sea exchange in rough and stormy seas measured by the dual-tracer technique. *Nature* **349**, 145-147.

Weissman, M.A., Ataktürk & Katsaros, K.B. 1984 Detection of breaking waves in a wind-generated wave field. *J. Phys. Oceanog.* **14**, 1608-1619.

Weller, R.A., Donelan, M.A., Briscoe, M.G. & Husug, N.E. 1991 Riding the crest - A tale of two wave experiments. *Bull. Am. Met. Soc.* **82**, 163-183.

Weller, R.A. & Farmer, D.M. 1992 Dynamics of the ocean mixed layer. *Oceanus* **35**, 46-55.

Wenz, G.M. 1962 Acoustic ambient noise in the ocean: spectra and sources. *J. Acoust. Soc. Am.* **34**, 1936-1956.

Wesely, M.L., Cook, D.R., Hart, R.L. & Williams, R.M. 1982 Air-sea exchange of CO_2 and evidence for enhanced upward fluxes., *J. Geophys. Res.* **87**, 8827-8832.

Wood, A.B. 1941 *A Textbook of Sound.*, G. Bell & Sons.

Woolf, D.K. & Thorpe, S.A. 1991 Bubbles and the air-sea exchange of gases in near-saturation conditions. *J. Marine Res.* **49**, 435-466.

Wu, J 1981 Bubble populations and spectra in near-surface ocean: summary and review of field measurements. *J. Geophys. Res.* **86**, 457-463.

Yoon, S.W., Crum, L.A., Prosperetti, A. & Lu, N.Q. 1991 An invstigation of the collective oscillations of a bubble cloud. *J. Acoust. Soc. Am.* **89**, 700-706.

Young, T.L. & Van Woert, M.L. 1989 *Plot88 software library reference manual,* Plotworks, Ramona, CA, 1989.

Zahn, M. 1979 *Electromagnetic Field Theory.* Krieger Publishing Co., Malabar, Fl.

Zedel, L. Farmer, D. 1991 Organized structures in subsurface bubble clouds: Langmuir circulation in the open ocean. *J. Geophys. Res.* **96**, 8889-8900.

Appendix A

Low-frequency sound-speed in a bubbly mixture: Wood's equation

This appendix contains the derivation of Wood's equation [Wood, 1941] for the speed of sound is a bubbly mixture as a function of void-fraction. The reflection coefficient for low-frequency acoustic waves propagating in a bubbly mixture is also given.

A.1 Sound-speed as a function of void-fraction

When a pressure wave propagates through a bubbly mixture, most of the compression is done on the easily compressible bubbles and very little on the fluid itself. The increase in the compressibility of the mixture and the negligible change in density (say for $\alpha<0.01$) will result in a reduced propagation speed. As for the attenuation of the pressure wave, it will be increased due to the increased damping caused by the bubbles. The acoustic resistance of the mixture $\rho_m c_m$ where ρ_m is the density of the mixture and c_m is the sound-speed in the mixture will also be reduced and therefore one might expect significant reflection from a pressure wave incident on a bubbly mixture with high void-fraction.

Wood [1941] arrived at these conclusions by assuming that the bubbly mixture was homogeneous with a bulk density and compressibility. This implicitly assumes that the frequency of the pressure wave is well below the resonant frequency of the largest bubbles in the mixture and that bubbles cannot interact. The former assumption guarantees that the dispersion due to the individual bubbles is negligible and this condition is usually more exacting than the latter in typical geophysical applications. (Appendix B will treat the dispersive effect caused by bubbles).

The following development is essentially that given by Wood [1941]. Assume a mixture composed of a constituent with density ρ_a and compressibility K_a dispersed in another constituent of density ρ_w and compressibility K_w. The proportion by volume of the first constituent is α and by volume conservation the proportion of the second constituent is $1-\alpha$. The bulk density of the mixture becomes

$$\rho_m = \alpha\rho_a + (1-\alpha)\rho_w \, , \qquad \text{(A.1)}$$

and the bulk compressibility is

$$K_m = \frac{\Delta V_m}{V_m \Delta P} = \frac{\Delta V_a + \Delta V_w}{V_m \Delta P} = \frac{\Delta V_a}{V_a \Delta P}\frac{V_a}{V_m} + \frac{\Delta V_w}{V_w \Delta P}\frac{V_w}{V_m} \qquad (A.2)$$

$$K_m = \alpha K_a + (1-\alpha) K_w . \qquad (A.3)$$

The sound-speed of the mixture can therefore be derived by using A.1 and A.3

$$c_m = \frac{1}{\sqrt{\rho_m K_m}} = \frac{1}{\sqrt{(\alpha\rho_a + (1-\alpha)\rho_w)\ (\alpha K_a + (1-\alpha)K_w)}} . \qquad (A.4)$$

The compressibility of an ideal gas depends on whether the heat exchange process between the bubble and its surrounding is adiabatic or isothermal [Karplus, 1958]. In other words, whether the heat exchange is slow or fast compared to a period of the pressure wave. The compressibility for each case is given by the adiabatic and isothermal gas laws respectively,

$$PV^{\gamma} = \text{const} , \quad PV = \text{const}$$

$$\frac{\Delta V}{\Delta P} = -\frac{V}{\gamma P}, \quad \frac{\Delta V}{\Delta P} = -\frac{V}{P},$$

$$K_{a,\,adi} = \frac{1}{\gamma P}, \quad K_{a,\,iso} = \frac{1}{P}, \qquad (A.5)$$

where γ is the specific heat ratio. Experiments conducted by Karplus [1958] showed that for air bubbles in water, the isothermal condition is the correct formulation and therefore the heat exchange between the bubble and its surrounding is rapid compared to one period of the pressure wave. Using the isothermal formulation for the compressibility and A.4 and noting that sound propagation in a single phase medium is adiabatic $c_a = 1/\sqrt{K_{a\,adi}\,\rho_a}$ we obtain the following expression for the sound-speed valid over the full range of void-fraction

$$\frac{1}{c^2} = \frac{\alpha^2\gamma}{c_a^{\,2}} + \frac{(1-\alpha)^2}{c_w^{\,2}} + \alpha(1-\alpha)\left(\frac{\rho_w}{P} + \rho_a K_w\right). \qquad (A.6)$$

At room temperature, typical values of the parameters in equation A.6 are [Karplus, 1958]

$c_a = 3.4 \times 10^4$ cm/s

$c_w = 1.5 \times 10^5$ cm/s

$\rho_a = 1.2 \times 10^{-3}$ gm/cm^3

$\rho_w = 1.0$ gm/cm^3

$P = 1.01 \times 10^6$ dyne/cm^2 (atmosphere)

$K_w = 4.54 \times 10^{-11}$ cm^2/dyne

$\gamma = 1.4$.

Figure A.1 shows how the sound-speed varies as a function of the void-fraction in an air-water mixture. Below a 10^{-2} void-fraction, the density (ρ_m) is relatively constant and all of the sound-speed reduction comes from the large increase in the compressibility of the mixture (K_m) due to the presence of bubbles. Above a 10^{-2} void-fraction, the compressibility is relatively constant because it is dominated by the air but the density of the mixture sharply decreases such that the sound-speed increases rapidly towards its value in air.

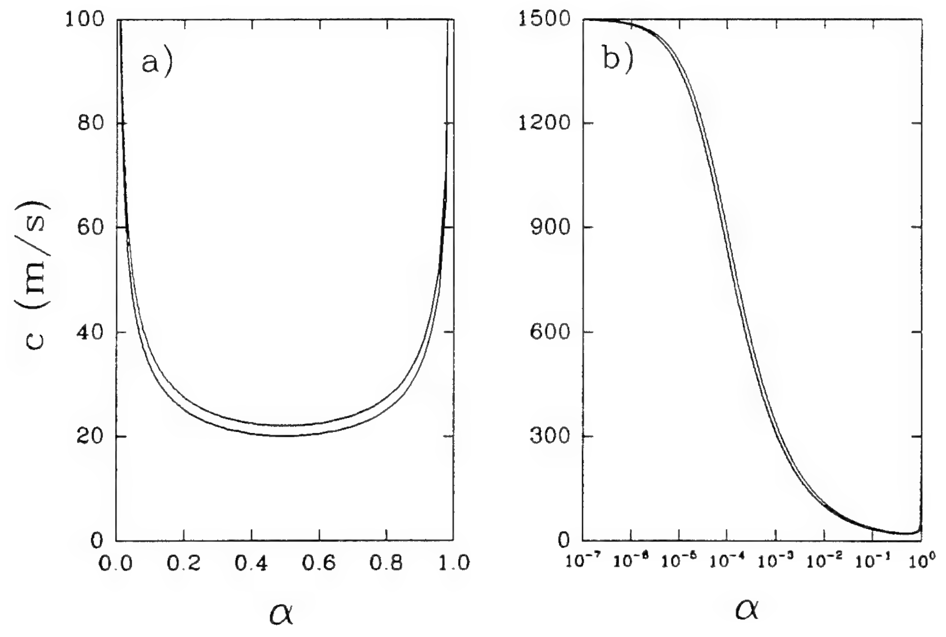


Figure A.1: Low frequency sound-speed as a function of void-fraction. a) linear-linear scale. b) log-linear scale. The lower curve on both plots is for atmospheric pressure and ambient temperature. The upper curve is for atmospheric pressure and a hydrostatic pressure corresponding to a depth of 2m and ambient temperature.

At void-fractions below 2×10^{-5}, equation A.6 can be approximated by the linear equation

$$c = 1500\mathrm{m/s} - \frac{160\mathrm{m/s}}{1.1\times10^{-5}}\alpha \qquad \text{(A.7)}$$

which yields sound-speeds accurate to ±5m/s over a range of $1500\mathrm{m/s} \geq c \geq 1340\mathrm{m/s}$. This linear approximation at low void-fractions is useful in comparing sound-speed and void-fraction measurements. Figure A.2 shows how equations A.6 and A.7 compare at low void-fractions.

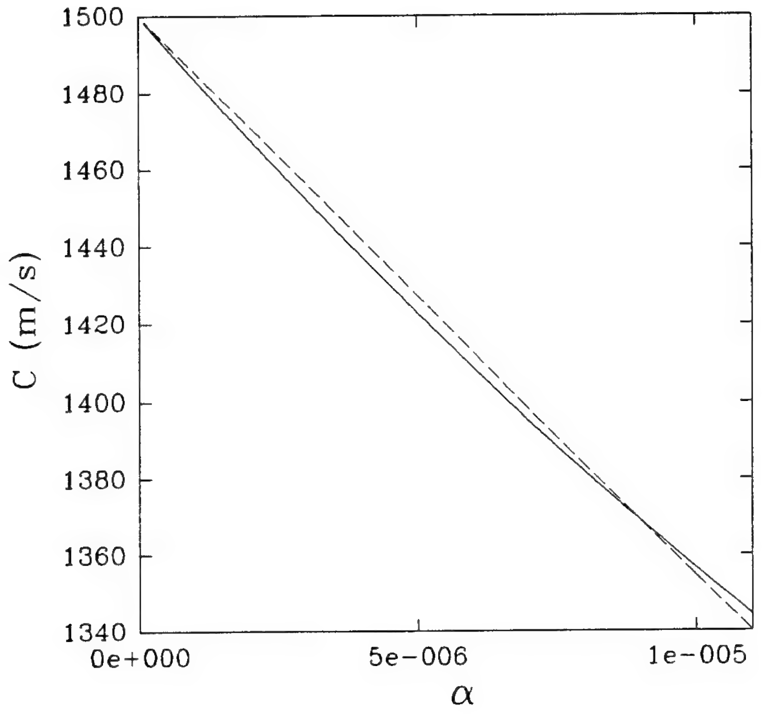


Figure A.2: Low frequency sound-speed as a function of void-fraction at very low void-fractions. The solid line is from the exact equation A.6 and the dashed line is from the linear approximation given by equation A.7. Note that the graph has linear-linear scales.

A.2 Reflection from a bubbly layer

It is interesting to calculate by how much a normally incident pressure wave will reflect as it propagates from a thick (greater than an acoustic wavelength) water layer to a thick bubbly layer. The solution is obtained by matching the velocity and the instantaneous pressure of the incident, reflected and transmitted pressure waves at the interface. The solution for the refection coefficient is given by Clay and Medwin [1977] as

$$R_{w,m} = \frac{\rho_w c_w - \rho_m c_m}{\rho_w c_w + \rho_m c_m} \tag{A.8}$$

where $R_{w,m}$ is the reflection coefficient for a propagation from water to bubbly mixture. The acoustic impedance for the mixture is $\rho_m c_m = \sqrt{\rho_m / K_m}$ where ρ_m and K_m are given by A.1 and A.3 respectively. For example, at void-fractions of 10^{-5}, 10^{-4}, 10^{-3} and 10^{-2} the fraction of energy reflected is 0.14%, 5.7%, 36% and 75% respectively.

Appendix B

Sound dispersion in a bubbly mixture

This appendix gives a brief derivation of the complex sound-speed (i.e. phase velocity and attenuation) of a pressure wave propagating in a bubbly mixture. Most of the work presented in this appendix is based on the book by Clay and Medwin [1977, appendix 6].

B.1 Bubble resonance

Because of their compressibility (stiffness^{-1}) and inertia (from the added fluid mass surrounding the bubble) bubbles in water can experience harmonic oscillations when excited by an external harmonic pressure forcing. As with any mechanical oscillator, damping is also present in the system and limits the amplitude of the pulsations. The equation of motion for the bubble is

$$m\ddot{\xi} + R_M\dot{\xi} + s\xi = 4\pi a^2 P_P e^{i\omega t} \qquad \text{(B.1)}$$

where

$m = 4\pi a^3 \rho_A$	effective mass of the bubble due to surrounding fluid,
ξ	radial displacement of the bubble surface,
$R_M = \omega m \delta$	mechanical damping constant,
$\omega = 2\pi f$	radian frequency,
δ	damping due to scatter, viscosity and thermal conductivity,
a	bubble radius,
$s = 12\pi\gamma P_A a$	stiffness constant,
P_A	ambient pressure,
γ	ratio of specific heats for enclosed gas,
P_P	amplitude of the pressure forcing.

Note that the derivatives ($\dot{}$) are with respect to time. The first and third term in B.1 are due to the inertia and stiffness of the bubble from which we can compute (by assuming harmonic oscillation) the natural frequency of oscillation of the bubble

$$f_R = \frac{1}{2\pi}\left(\frac{s}{m}\right)^{1/2} = \frac{1}{2\pi a}\left(\frac{3\gamma P_A}{\rho_A}\right)^{1/2} \qquad \text{(B.2)}$$

where the subscript R refers to resonance. This result is based on the assumption that the surface tension is negligible and the bubble pulsates adiabatically (heat exchange between bubble and surrounding water is slow compared to one period of the excitation). For air bubbles in water the equation is simplified to

$$f_R(\text{kHz}) = \frac{3.25}{a(\text{mm})}\sqrt{1 + 0.1z} \qquad \text{(B.3)}$$

where z is the depth in meters. It can be shown that for radii greater than 2μm the error due to the above assumptions will be less than 8% [Clay and Medwin, 1977]. An interesting result from the above equation is that $a_R k_R$=0.0136 (at the surface). Hence, the resonant acoustic wavelength is much longer than the bubble radius. The solution of B.1 for the radial displacement is given by

$$\xi = \frac{-i4\pi a^2\,(P_p/\omega)\,e^{i\omega t}}{R_M + i\,(\omega m - s/\omega)}\ . \qquad \text{(B.4)}$$

B.2 The complex sound-speed

A pressure wave becomes dispersive (i.e. speed of propagation is a function of frequency) when it travels through a bubbly mixture at a frequency near the resonance frequency of the ambient bubbles. The formulation for the dependance of the sound-speed on frequency and bubble density is given next.

The sound-speed in the mixture is given by

$$c_m = \frac{1}{\sqrt{\rho_m K_m}} \qquad \text{(B.5)}$$

where K_m and ρ_w are the density and compressibility of the mixture. For typical oceanographic applications, the void-fraction is usually small (say α<0.01) and therefore the density of the mixture is essentially the density of water. In this case, all changes in the speed of sound are due to important changes in the compressibility which occur even at small void-fraction. Of course, the assumption that the density does not contribute is not valid immediately under a breaking wave where void-fraction above 10% have been observed [§2,3,4 and 5]. The compressibility K_m is composed of two parts

$$K_m = K_w + K_a \tag{B.6}$$

where $K_w = 1/\sqrt{\rho_w\, c_w}$ is due to the compressibility of bubble-free water and K_a is due the compressibility of the bubbles and it is given by

$$K_a = \frac{\Delta V/V}{\Delta P} = \frac{N\Delta v}{\Delta P} = \frac{NS\xi}{P_P e^{i\omega t}} = \frac{NS^2}{m\omega^2\,[(-1 + \omega_R{}^2/\omega^2) + i\,R_M/(\omega m)]} \tag{B.7}$$

where N is the number of bubbles per unit volume, Δv is the change in volume of each bubble, ξ is the radial displacement of the bubble as given by B.4 and S is the surface area of the bubble ($4\pi a^2$). It has been assumed, for the moment, that the bubbles are of equal size; generalization to a mixture with different sizes will be made later. We can simplify the notation by defining the frequency ratio $Y = f_R/f$ and $\delta = R_M/(\omega m)$ (from B.1). Then the compressibility K_a becomes

$$K_a = \frac{N\,4\pi a\,[Y^2 - 1 - i\delta]}{\rho_w\omega_2\,[(Y^2 - 1)^2 + \delta^2]}\,. \tag{B.8}$$

Upon substitution of K_a and K_w into B.5 yields

$$c = \frac{1}{\sqrt{\rho_m\,K_m}} = \frac{c_w}{\sqrt{1 + A - i\,B}} \tag{B.9}$$

where

$$A = \frac{Y^2 - 1}{(Y^2 - 1)^2 + \delta^2}\;\frac{4\pi a\,N\,c_w{}^2}{\omega^2}, \quad B = \frac{\delta}{(Y^2 - 1)^2 + \delta^2}\;\frac{4\pi a\,N\,c_w{}^2}{\omega^2}\,. \tag{B.10}$$

The interpretation of the complex sound-speed becomes clearer if we consider the complex wavenumber. The equation for a plane pressure wave propagating in a bubbly mixture is given by

$$p_P = P_P\,e^{-k_{im}\,x}\;e^{i(\omega t\,-\,k_{re}\,x)} \tag{B.11}$$

where the propagation wave number k_{re} and the attenuation k_{im} are the real and imaginary part of the complex wavenumber k

$$k = \frac{\omega}{c} = \frac{\omega\sqrt{1 + A - iB}}{c_w} \tag{B.12}$$

where the subscript w refers to bubble free water. In typical oceanic conditions, A and B are very small [Clay and Medwin, 1977] and we can approximate B.12 by taking the Taylor expansion of the term inside the square root.

$$k \cong k_w \left(1 + \frac{A}{2} - \frac{iB}{2}\right) \tag{B.13}$$

where $k_w = \omega/c_w$. The real and imaginary parts of k are given by

$$k_{re} = k_w \left(1 + \frac{A}{2}\right), \quad k_{im} = k_w \frac{B}{2}. \tag{B.14}$$

Substituting B.10 into B.14 we obtain the final form of the real and imaginary wavenumbers for a bubble population of uniform size

$$k_{re} = k_w \left(1 + \frac{2\pi a N c_w^2}{\omega^2} \frac{Y^2 - 1}{(Y^2 - 1)^2 + \delta^2}\right) \tag{B.15}$$

$$k_{im} = k_w \left(\frac{2\pi a N c_w^2}{\omega^2} \frac{\delta}{(Y^2 - 1)^2 + \delta^2}\right) \tag{B.16}$$

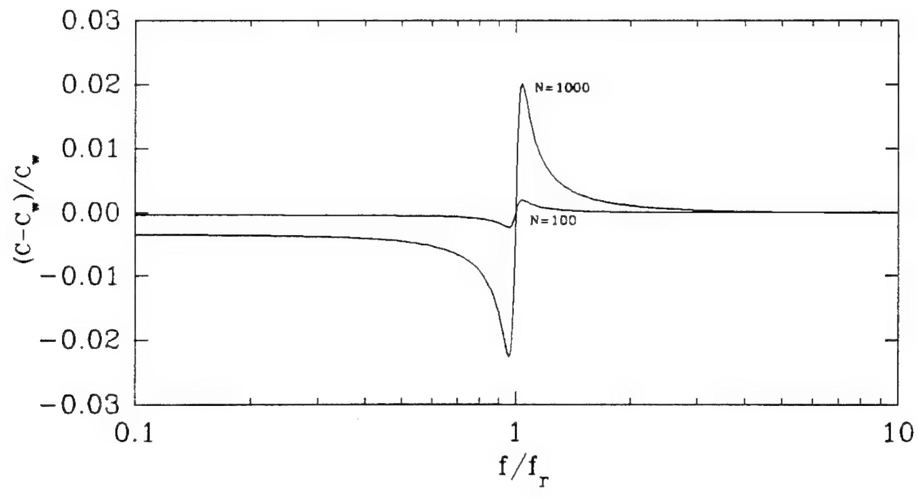


Figure B.1: Dispersion of sound phase velocity for a uniform bubble population. The frequency f is normalized by the resonance frequency of the bubbles f_R. The sound-speed anomaly $\Delta c = c - c_w$ is normalized by the sound-speed in bubble free water c_w. The data shown is for bubbles of size 100μm and a number of bubbles per m^3 N=100 and N=1000. The low-frequency asymptote is a function of the void-fraction and the high-frequency asymptote is the speed of sound in bubble-free water.

Figure B.1 shows the effect of dispersion on the sound velocity for bubbles of uniform size. Note that the sound velocity is $c_{re} = \omega/k = \omega k_{re}/(k_{re}^2 + k_{im}^2)$ but since A and B are typically very small compared to unity this implies that $k_{im} << k_{re}$ and thus the sound velocity can be approximated by $c_{re} \equiv \omega/k_{re}$. Note that the high frequency asymptote is equal to the sound-speed in bubble free water and the low frequency asymptote is a function of the void-fraction only. The greater the number of bubbles per unit volume N, the more important the dispersion becomes. The effect of a mixture of bubble sizes is to smear the dispersion curve since at a given frequency, bubbles with smaller resonance frequencies will increase the sound velocity and bubbles with higher resonance frequencies will compensate by decreasing the sound velocity.

Equation B.15 and B.16 can be generalized to a random distribution of bubbles by replacing N with n(a)da where n(a) is the number of bubbles per unit volume in the radius increment between a and a+da. Integration over all radii gives

$$k_{re} = k_w \left(1 + 2\pi \int_a \frac{a^3\, n(a)}{a^2 k_R^2} \frac{Y^2(Y^2 - 1)}{[(Y^2 - 1)^2 + \delta^2]}\, da \right) \tag{B.17}$$

$$k_{im} = k_w \left(2\pi \int_a \frac{a^3\, n(a)}{a^2 k_R^2} \frac{Y^2 \delta}{[(Y^2 - 1)^2 + \delta^2]}\, da \right) \tag{B.18}$$

where $ak_R = 0.0136$ is a constant for air bubbles at sea level (from B.2). We can now give the expression for the phase velocity

$$\mathrm{Re}\{c_m\} \equiv \frac{\omega}{k_{re}} \equiv c_w \left(1 - 2\pi \int_a \frac{a^3 n(a)}{a^2 k_R^2} \frac{Y^2(Y^2 - 1)}{[(Y^2 - 1)^2 + \delta^2]}\, da \right) \tag{B.19}$$

where we have used the leading order Taylor expansion of the fraction ω/k_{re}. Thus, integrating equations B.18 and B.19 (or B.17) yields the attenuation and the phase velocity of an acoustic wave propagating at a frequency f. Insight into the behavior of these integrals can be gained by looking more specifically at their kernel

$$K_{re} = \frac{Y^2(Y^2 - 1)}{[(Y^2 - 1)^2 + \delta^2]} \tag{B.20}$$

$$\mathcal{K}_{im} = \frac{Y^2\delta}{[(Y^2 - 1)^2 + \delta^2]} \tag{B.21}$$

where $\mathcal{K}_{re}$ and $\mathcal{K}_{im}$ refer to the kernels in equations B.18 and B.19 (or B.17) respectively. Figure B.2 shows the shape of the kernels for three distinct bubble resonance frequencies (f_R=1, 10 and 100kHz). At frequencies much below bubble resonance $\mathcal{K}_{re}$=1 and the integral reduces to computing the void-fraction. At frequencies much above bubble resonance, $\mathcal{K}_{re}$=0 and hence bubbles in that range do not contribute to the phase velocity. Figure B.2b shows that only bubbles excited at or close to their resonance frequencies contribute to the attenuation since $\mathcal{K}_{im}$=0 away from resonance. The different amplitudes of the curves in figure B.2a,b are due to the damping δ which varies as a function of frequency.

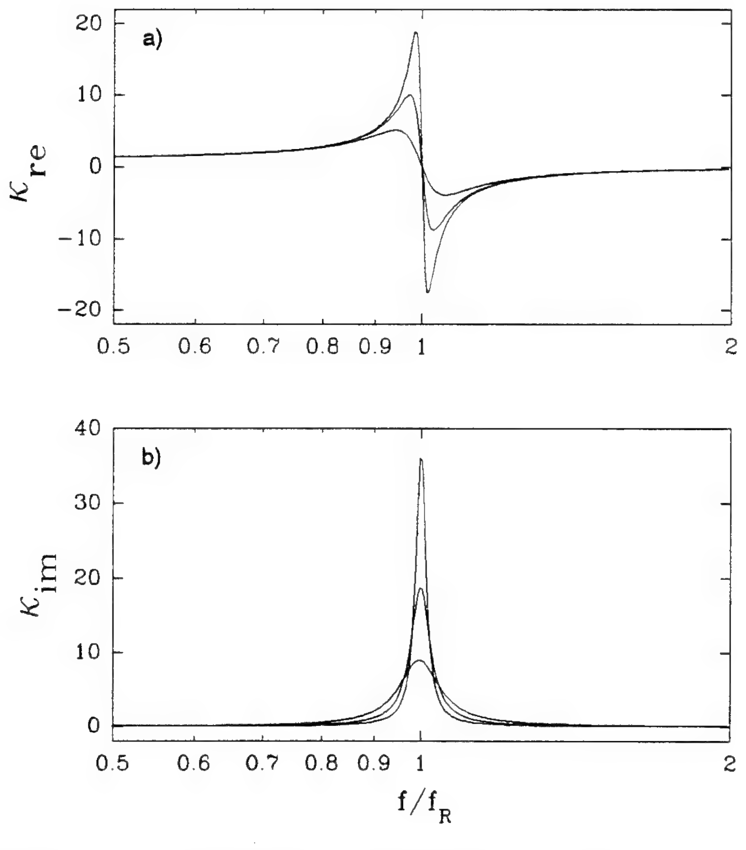


Figure B.2: Plot of the kernels of equations B.18 and B.19 (or B.17). The upper, middle and lower curves on each plot are for bubbles of radius 3250μm, 325μm and 32.5μm (f_R=1kHz, 10kHz and 100kHz resonance frequencies) respectively. At frequencies much below bubble resonance $\mathcal{K}_{re}$=1 while much above bubble resonance $\mathcal{K}_{re}$=0. $\mathcal{K}_{im}$ is zero both above and below bubble resonance. Dispersion and attenuation effects are generated at or near bubble resonance.

Two special cases of the above expression are the low frequency and high frequency approximations. At low frequency, $f \ll f_R$ and the sound-speed becomes

$$c_w^{lf} = c_w \left(1 - \frac{3\alpha}{2\, a^2 k_R^2}\right) \tag{B.22}$$

where

$$\alpha = \int_a \frac{4}{3}\pi a^3\, n(a)da \tag{B.23}$$

is the void-fraction of the mixture. At frequencies well below bubble resonance, the sound-speed is therefore no longer dispersive. It must be pointed out that this low frequency result is based on an adiabatic heat exchange process between the bubble and the surrounding fluid. This assumption is valid when the exchange is slow compared to one wave period of the pressure wave and clearly this assumption is violated at sufficiently low frequencies where the exchange is more nearly isothermal (rapid) compared to a period of the pressure wave. Furthermore, we assumed that the sound-speed fluctuations due to density differences were negligible. Obviously, this assumption fails when the void-fraction becomes large (above 1%). When comparing equation B.22 with equation A.7 which does not suffer from the above restrictions one finds that they agree to within 1% for void-fractions below 10^{-5}.

At high frequency $f \gg f_R$, the sound-speed becomes

$$c_w^{lf} = c_w \left(1 + \frac{3\alpha Y^2}{2\, a^2 k_R^2\, (1 + \delta^2)}\right) \tag{B.24}$$

which tends toward c_w at high enough frequencies.

Appendix C
Generation of the breaking wave packet

This appendix describes the generation of the breaking wave packet for the 2D experiment of §3. Details on the measurement of the wavemaker transfer function and wave tank settling time are also given. Finally, the relationship between dissipation and wave packet slope is given along with some comments on the actual measurement of dissipation.

C.1 Generation of the breaking wave packet

The technique for generating the breaking wave packet is described at length in Rapp and Melville [1990]. The formulation of the method presented here is a summary of their work and is included tor completeness.

The free surface displacement at a time t and position x along the wave channel can be specified as a summation of N sinoisoidal wave components

$$\eta(x,t) = \sum_{n=1}^{N} a_n \cos(k_n x - 2\pi f_n t - \phi_n) \qquad \text{(C.25)}$$

where a_n, k_n, f_n and ϕ_n are the amplitude, wavenumber, frequency and phase of the nth wave component. The frequency and the wavenumber are related by the linear dispersion relation

$$\frac{(2\pi f_n)^2}{g} = k_n \tanh(k_n h) \qquad \text{(C.26)}$$

where g is the graviational constant and h is the depth of water in the channel. The actual number of components N is not very critical as long as it is large enough to approximate a smooth spectrum. In this study, 32 components were spread evenly over a frequency bandwidth $\Delta f = f_N - f_1 = 0.88$Hz. The center frequency of the wave packet is defined as $f_c = (f_1+f_N)/2$ and for these experiments it was set at 0.88Hz.

The phase of each component ϕ_n is adjusted such that all the wave crests of the N components arrive simultaneously at a point x_b in the wave channel. This is done by setting

$$\cos(k_n x_b - 2\pi f_n t_b - \phi_n) = 1 \tag{C.27}$$

where t_b is the time when breaking occurs. With the above, the surface displacement becomes

$$\eta(x,t) = \sum_{n=1}^{N} a_n \cos[k_n(x-x_b) - 2\pi f_n(t-t_b)]. \tag{C.28}$$

The mean position of the wavemaker is at x=0 and therefore the surface displacement at the paddle is

$$\eta(x,t) = \sum_{n=1}^{N} a_n \cos[-k_n x_b - 2\pi f_n t']). \tag{C.29}$$

where $t'=t-t_b$. The effect of changing t_b is to advance or delay the wave packet without changing its waveform or the theoretical breaking location x_b.

In the present study, the amplitude of each component a_n was adjusted such that the slope $a_n k_n$ of each of the component remained constant. The wave packet can be characterized in terms of a slope parameter S

$$S = G\, N\, a_n\, k_n \tag{C.30}$$

where G is the gain factor used to increase or decrease the slope of the packet. Figure C.1 shows the computed wave packet used in the 2D experiment of §3.

C.2 Wavemaker transfer function

Because of the inherent inertia and damping of the wavemaker system the amplitude and phase of the input waveform must be corrected for the transfer function of the wavemaker system in order to obtain the desired surface displacement. The transfer function was measured by generating a burst of small amplitude sinuisoidal waves (O(10) waves) and by measuring the resulting surface displacement 2m downstream of the paddle. The transients at the beginning and at the end of the burst were excluded from the measurements. Only the center section of the burst where the waves had constant amplitude and phase was considered. The waves within the burst had amplitude no greater than 1cm. Small amplitude waves were used because the wave packet formulation is based on linear theory.

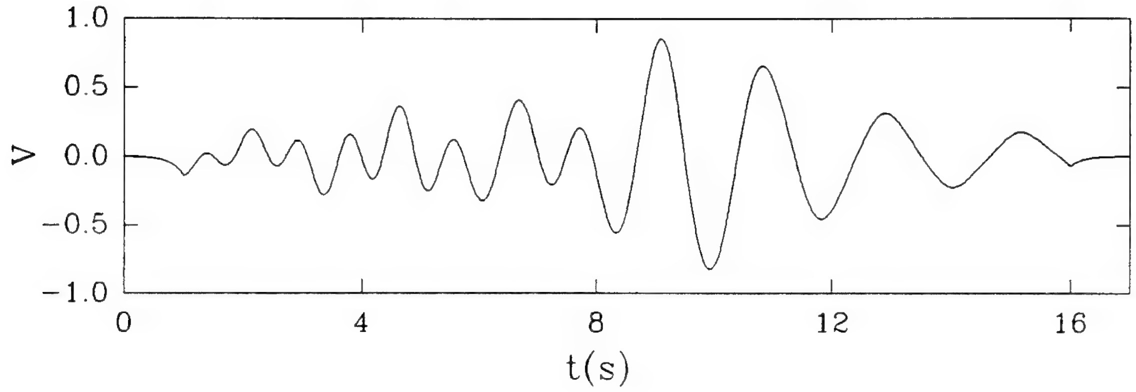


Figure C.1: Computed breaking wave packet used in the 2D experiment of §3. High frequency waves are generated first followed by lower frequency waves. Dispersion ensures superposition at a predetermined position x_b from the wave paddle. A PC equipped with a Metrabyte Das20 board performed the D/A at a conversion rate of 100Hz. The analog signal was fed to the input of the wavemaker system. Note that the packet shown has been corrected for the wavemaker transfer function.

Figure C.2 shows the measured amplitude and phase transfer functions. The solid line in the amplitude transfer function is a third order fit

$$T(\text{cm/V}) = -2.72 + 21.18f - 17.13f^2 + 4.05f^3 ,$$

where f is in Hz. The solid line in the phase transfer function is a linear fit

$$\delta\phi\ (\text{rad}) = 1.14 - 1.13f .$$

Figure C.2 shows that for an input frequency of 1Hz and an input amplitude of 1V, the surface displacement amplitude 2m in front of the paddle will have an amplitude of ~5.5cm and zero phase shift with respect to the input waveform. The amplitude transfer function is typical of piston type wavemakers [Dean and Dalrymple, 1984]. The initial rise with frequency up to 1Hz shows that the wavemaker more or less keeps up a constant stroke. For a constant stroke, the same volume of water is displaced by the wave paddle. Hence, as the frequency of the waves increases, we expect their amplitude to increase as well according to

$$\frac{H}{S} = kh$$

where H is the wave height, S is the full stroke, h is the water depth and k is the wavenumber [Dean and Dalrymple, 1984]. After 1Hz, the hydraulic system cannot move the wave paddle through its full stroke anymore and hence the amplitude of the waves diminishes.

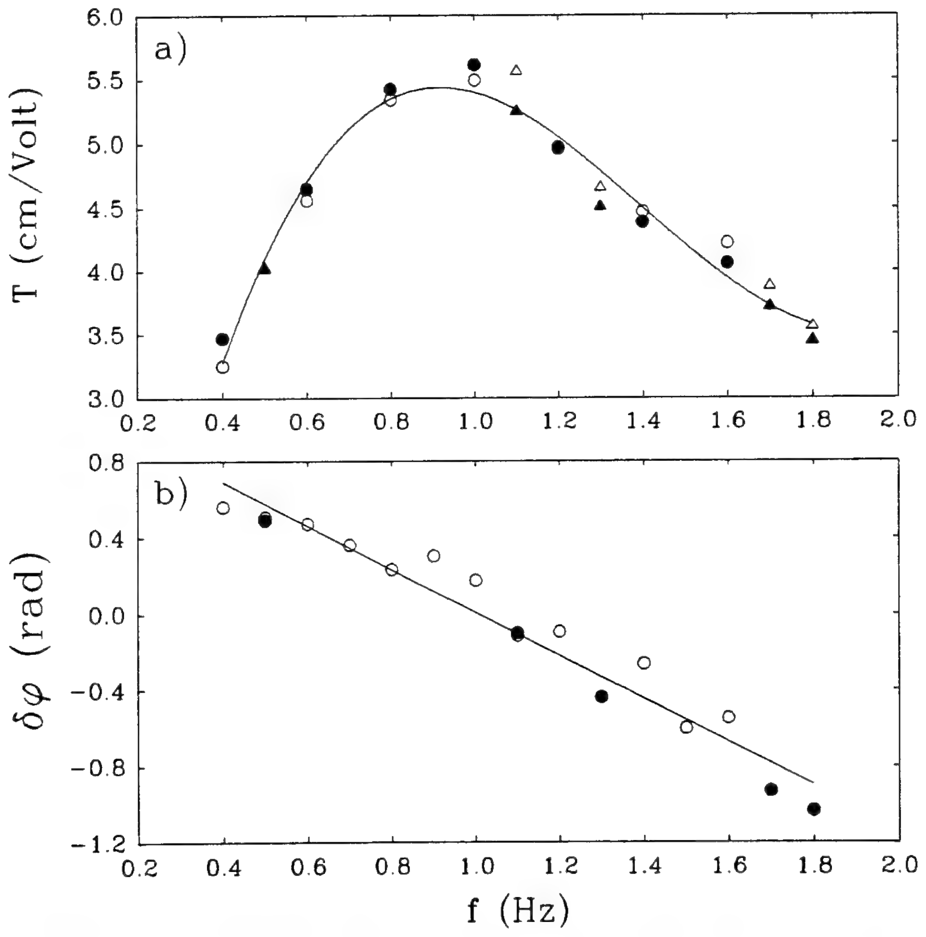


Figure C.2: a) Amplitude and b) phase transfer function for the MIT wavemaker system installed on the wave channel 25m × 0.7m ×0.6m. The hollow symbols are for an input amplitude of 0.1V and the filled symbols are for an input amplitude of 0.2V. Transfer function measured on 08-24-90.

Ideally, the amplitude and phase transfer functions would be flat in the frequency range of interest in which case no corrections to the input waveform would be necessary. Figure C.2 shows that this is clearly not the case. It is therefore necessary to correct each of the 32 components in the wave packet in the following manner

$$\eta(x,t) = \sum_{n=1}^{N} \frac{a_n}{T_n} \cos[-k_n x_b - 2\pi f_n t' - \delta\phi]). \qquad \text{(C.31)}$$

where a_n is divided by T_n and the phase shift $\delta\phi$ is substracted from the phase of the signal. When the corrected wave packet is applied to the wavemaker system, each frequency component is influenced by the transfer function. This will essentially translate into multiplying the amplitude by T_n and adding $\delta\phi$ to the phase and hence the desired surface displacement will be recovered.

C.3 Wave channel settling time

In order to obtain good repeatability in the wave profile and bubble plume generated by laboratory breaking waves, the wave channel must be allowed to rest in between each repetition of the breaking wave event.

A simple test was conducted in order to establish a reasonable waiting period. The standard deviation of the surface displacement was measured over 20s windows from immediately after breaking up to 16min after breaking. The results are shown in figure C.3 where the standard deviation of the surface displacement is plotted as a function of time. There are five different symbols shown corresponding to five repeats of the experiment. Clearly, the wave channel quickly reaches a calm state with a minimum standard deviation of 0.05mm approximately 6min after onset of breaking. This minimum is probably the noise level of the wave gauge electronics. The data points that are off the curve at ~8min and ~12min were caused by the subway which has its underground line running close to the laboratory. A 6min wait time between each repetition of the breaking event was used for the 2D experiment described in §3.

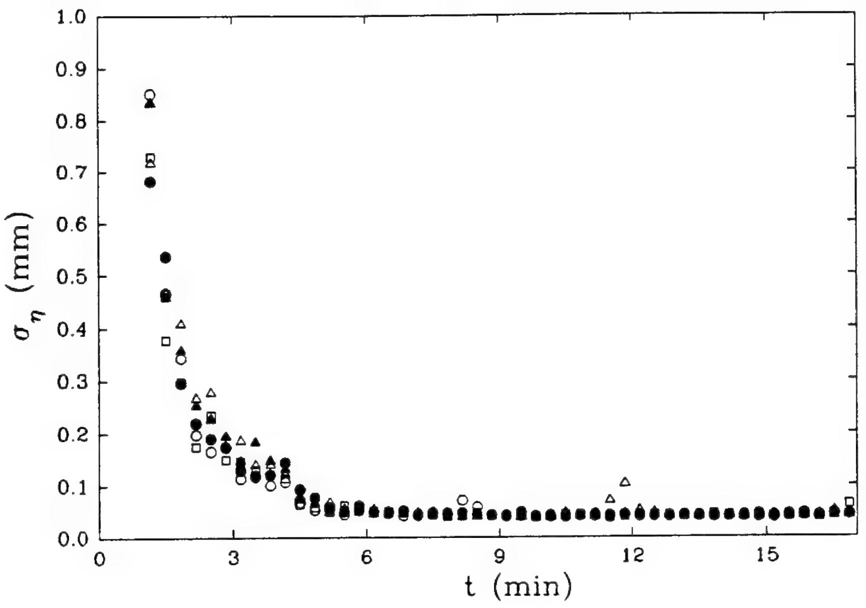


Figure C.3: Wave channel settling time. The standard deviation of wave gauge displacement measurement computed over 20s windows is shown as a function of the time after onset of breaking. The channel reaches a steady calm state approximately 6min after onset of breaking. The 5 different symbols represent 5 different repetitions of the breaking event.

C.4 Dissipation as a function of wave packet slope

The dissipation of wave surface energy caused by breaking can be estimated by taking time integrated free surface displacement variance upstream and downstream of breaking. The free surface displacement variance $\overline{\eta^2}$ is given by

$$\overline{\eta^2} = \frac{1}{T}\int_0^T \eta^2(t)\, dt$$

where $\eta(t)$ is the surface displacement measured with wave gauges and T is the length of the sampling interval (40s). The dissipation is given by

$$D = \frac{\overline{\eta_o^2} - \overline{\eta_f^2}}{\overline{\eta_o^2}}$$

where $\overline{\eta_o^2}$ and $\overline{\eta_f^2}$ are the upstream and downstream free surface displacement variance. Figure C.4 shows the dissipation D as a function of the wave slope parameter S. The data for $S<0.35$ are for a wave packet not steep enough to initiate breaking. In this case, the dissipation is $D\cong 0.06$ which is entirely due to bottom and side walls viscous dissipation. For $S>0.35$, breaking increases in strength with the slope parameter S. The dissipation caused by breaking only D_b is computed by subracting the viscous dissipation from the measured dissipation: $D_b=D-0.06$. The three wave packet amplitudes studied in §3 had a slope S=0.38, 0.45 and 0.54 repectively. This range of slopes covers a significant fraction of the breaking regime showed in figure C.4. The plateau region for $S>0.6$ is caused by multiple breaking events occuring within the same wave packet [Rapp and Melville, 1990].

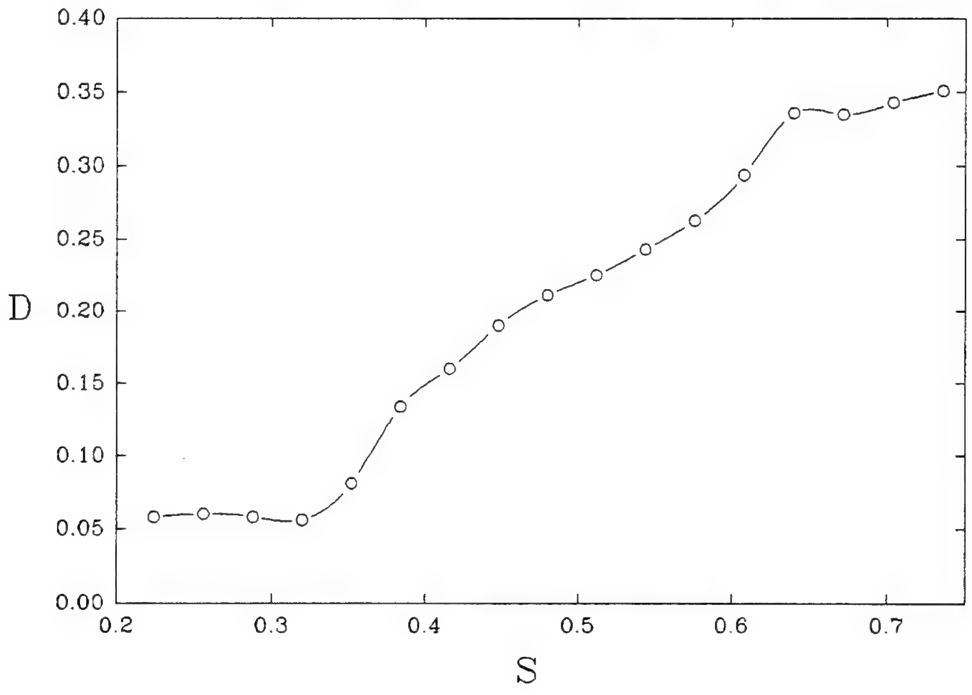


Figure C.4: The fractional dissipation of the wave packet energy as a function of the slope parameter S. The data shown is for the wave packet with center frequency f_c=0.88Hz, bandwidth Δf=0.88Hz and normalized breaking location $x_b k_c$=24.6 where k_c is the wavenumber of the center component of frequency f_c.

Appendix D
Effect of sampling rate on estimates of the time-delay

Recall that the peak of the cross-correlation function between the transmit and the receive acoustic pulses yields the time-delay for the direct path between the transmitter and the receiver (§6). From the time-delay, one can infer the speed of sound since the separation between the hydrophones is known and constant. In this appendix, we investigate the effect of sampling the original pulses at a higher or lower sampling rate. In particular, we investigate if higher sampling rates necessarily give increases in the resolution of the time-delay.

D.1 The model

The cross-correlation of two quasi-sinusoidal signals yields a function that is also quasi-sinusoidal. A simple model of the cross-correlation function in the neighborhood of its peak was used to investigate the effect of sampling rate.

We use half a period of a sinusoid to model the cross-correlation peak (see figure D.1). The equation of the model is

$$y(\tau) = \cos[\tau + \phi_n(\tau) + \phi_s] + A_m(\tau) \qquad \text{(D.1)}$$

where τ is the time-delay, $\phi_n(\tau)$ is random noise on the phase of the signal and it is distributed in the range $-n\pi<\phi_n(\tau)<n\pi$ where n is adjustable. The parameter ϕ_s is the initial starting phase of the first data point on the cross-correlation function (see "first sample" on figure D.1). It is randomly selected within the range $0<\phi_s<\pi/4$. $A_m(\tau)$ is the amplitude noise of the cross-correlation function and its range is $-m<A_m(\tau)<m$.

The number of data points on the cross-correlation function (figure D.1) is dependent on the sampling rate. For example, a 10kHz pulse sampled on both the transmit and receive signal at 500kHz would yield 25 points on the half-period cosine model of the cross-correlation peak (or 50 points per full period). The true location of the peak is always $\tau=0$. Because of noise and random starting phase, the location of the peak of the cross-correlation function will only be an estimate of the true peak (located at zero). The dark symbols in figure D.1 represent the data used to interpolate the peak between the data samples. These points are selected uniformly around the coarse peak (i.e. the peak of the cross-correlation function without interpolation).

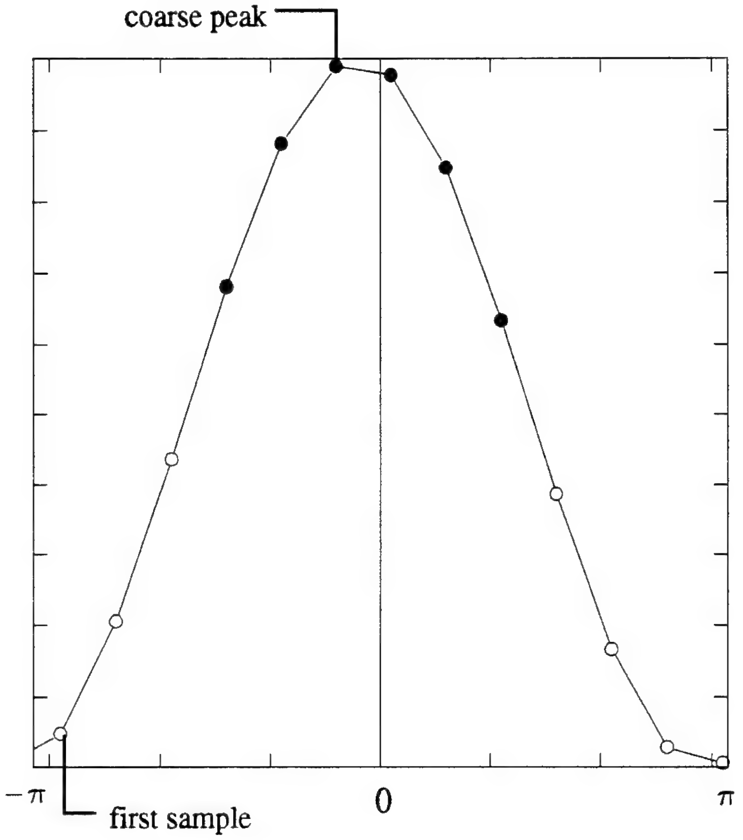


Figure D.1: Model used in the study of the effects of sampling rate and interploation (discussed in appendix E). The peak of the cross-correlation function is modelled as a half cosine. The symbols represent the actual samples which define the cross-correlation peak. The solid symbols are the data selected for the spline fit (i.e. for interpolation)

D.2 Optimum sampling rate as a function of noise

We first examined the effect of sampling rate. In the simulations that we have conducted, we selected a certain noise level for both n and m and then we proceeded with several simulations. For each simulation, a new value of ϕ_s was randomly selected. For each data point, a certain amount of phase and amplitude noise is added based on the chosen noise levels. A simulation consists of tracking the location of the peak of the cross-correlation (i.e. of the half-cosine model) and comparing it with the real peak located at zero. When ϕ_s=0 and the noise levels n=0 and m=0, the location of the peak of the cross-correlation is equal to the true location since for these conditions, a data sample falls exactly on the zero line. Of course, this solution is optimal but not repeatable from one simulation to the next since ϕ_s, n and m will most often take values different than zero. For this reason, the data that will be reported for the simulations are for worse case scenarios. Only the simulations giving the greatest deviation from zero will be reported.

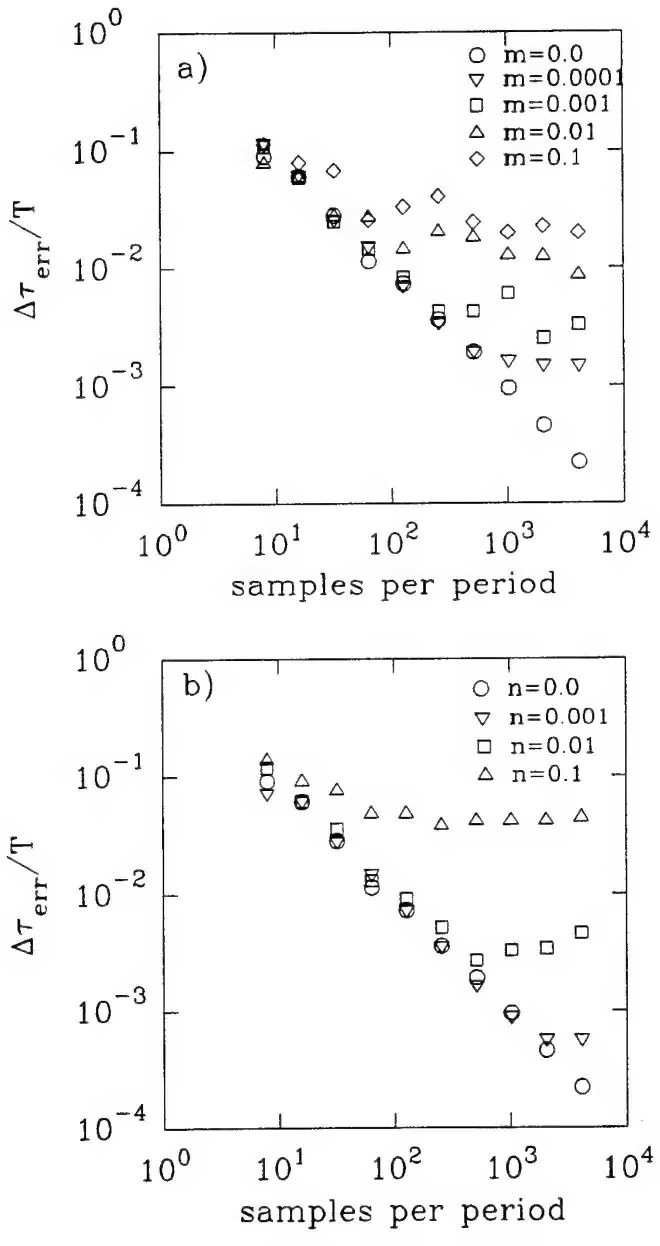


Figure D.2: Error $\Delta\tau_{err}$ on the resolution of the location of the peak of the cross-correlation function normalized by the acoustic wave period T. The sampling rate is expressed as the number of data samples per period. a) Amplitude noise A_m and b) phase noise ϕ_n. A 5 point spline interpolation was used in all simulations.

Figure D.2a shows the results from several simulations with no phase noise ϕ_n and various levels of amplitude noise A_m. The x-axis is the number of samples per acoustic period (i.e. the sampling frequency normalized by the acoustic frequency). The y-axis is the error $\Delta\tau_{err}$ in locating the true location of the peak normalized by the period of the acoustic wave. Hence, this error ratio represents a fraction of the acoustic period. When no noise is present, we would expect the error to steadily decrease with an increase in the sampling rate since more data points will help track the peak better. When noise is added to the simulations, a noise dependent cutoff appears beyond which there can be no further improvements in the resolution of the location of the peak no matter how fast the sampling rate is. The same conclusion holds true for simulations involving only phase noise (figure D.3b).

Hence, the optimal sampling rate is a strong function of the noise level present on the cross-correlation function. Beyond the optimal sampling rate, there is no further increases in the resolution of the time-delay with increases in the sampling rate.

Appendix E
Interpolation of the cross-correlation function

Using the same model as the one presented in appendix D, we can investigate the interpolation technique used to increase the resolution of the time-delay estimates. The basic questions motivating this study are "when does the interpolation technique ceases to give increases in the time-delay resolution" and "could interpolation compensate for a lower sampling rate"?

E.1 Effect of noise on the interpolation

When discussing interpolation issues, two parameters have to be kept in mind. First, the number of points used for the interpolation is effectively the number of data samples located at the peak of the cross-correlation function which are used in the cubic spline. This number is typically smaller than 20 for the present work. In general, the greater the number of points in the spline, the greater is the accuracy of the interpolation and the more time consuming is the computation of the spline. The second parameter is the interpolation factor which refers to the size of the finer mesh onto which interpolation is being performed. For example, a factor of ten would mean that the mesh is ten times finer or effectively the resolution is ten times greater.

With the help of the model described in appendix D, we proceed to examine several aspects of the interpolation technique. Figure E.1 shows how the error on the estimates of the time delay $\Delta\tau_{err}$ decreases as the number of interpolations is increased. For these simulations, we have set the amplitude (A_m) and phase (ϕ_n) noise to zero. The x-axis is the interpolation factor or the increase in resolution and the y-axis is the upper bound error on the estimate of the time-delay ($\Delta\tau_{err}$) normalized by the acoustic wave period T.

The most important feature of this plot is that all curves eventually level off at some high enough interpolation factor. This means that even for noise-free conditions, there will always exist an upper cutoff in the interpolation factor beyond which there is no more to be gained by interpolating further. However, notice that too much interpolation will never degrade the resolution; it will only require more computing time. The curves with the circles and the triangles have the same number of points in their spline but the latter has a sampling rate 5 times greater. The effect of increasing the sampling rate by a factor of 5 causes the error to drop by the same amount at an interpolation factor of 1 (i.e. no interpolation). Also notice that the curve with the higher sampling rate reaches a plateau at a higher interpolation factor. This is due to the fact that the spline is a local analysis. In the absence of noise, the spline performs much better if the input data points are located as close as possible to the peak of the cross-correlation. Since 5 points are selected for inputs to the spline in each case, the curve with the higher sampling rate will necessarily have its 5 points closer to the peak. Figure E.1 also shows that for a fixed sampling rate, the error will be smaller if more points are included in the spline. This effect seems to be more noticable at a high interpolation factor.

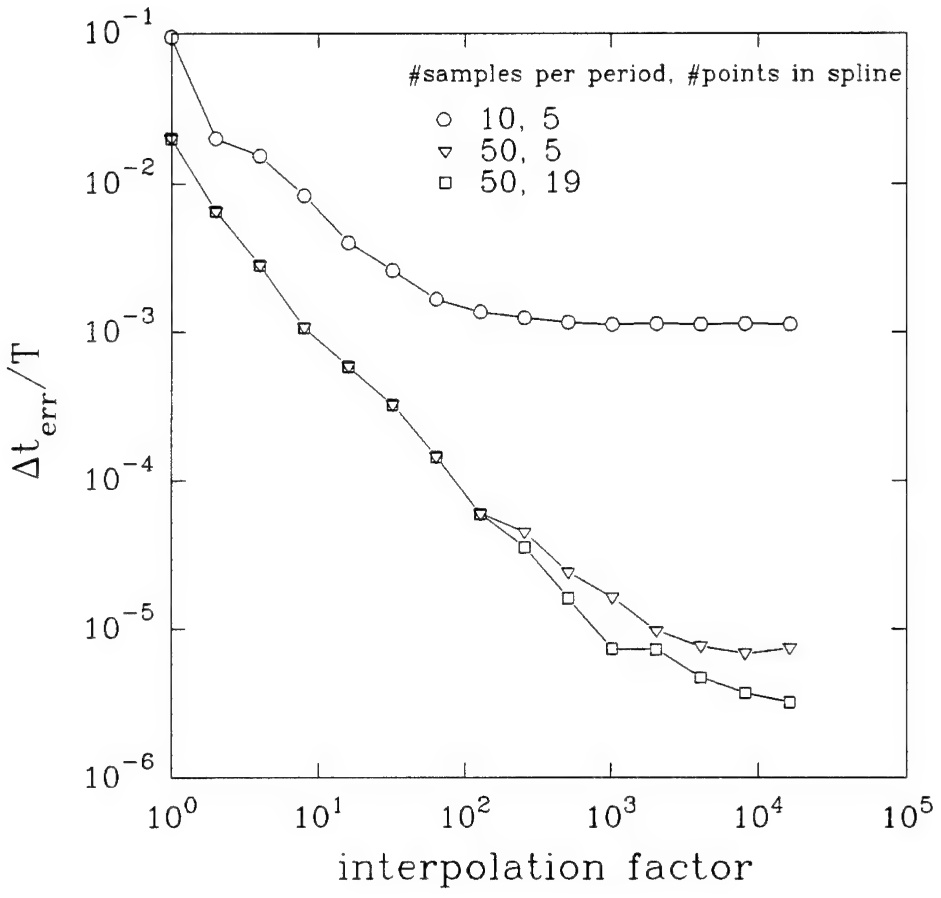


Figure E.1: Error $\Delta\tau_{err}/T$ in the resolution of the location of the peak of the cross-correlation function versus the interpolation factor. T is the acoustic wave period. The number of samples per period and the number of points used in the spline are shown. The sampling rate is expressed as the number of data samples per period.

We now look into the effect of adding amplitude and phase noise. Figure E.2 shows that with increasing noise, either amplitude or phase noise, the plateau region will begin at a lower interpolation factor. Hence, the amount of useful interpolation that can be performed is a strong function of the amount of noise on the cross-correlation function. This, in turn, is a function of the noise level on the original acoustic pulse time-series.

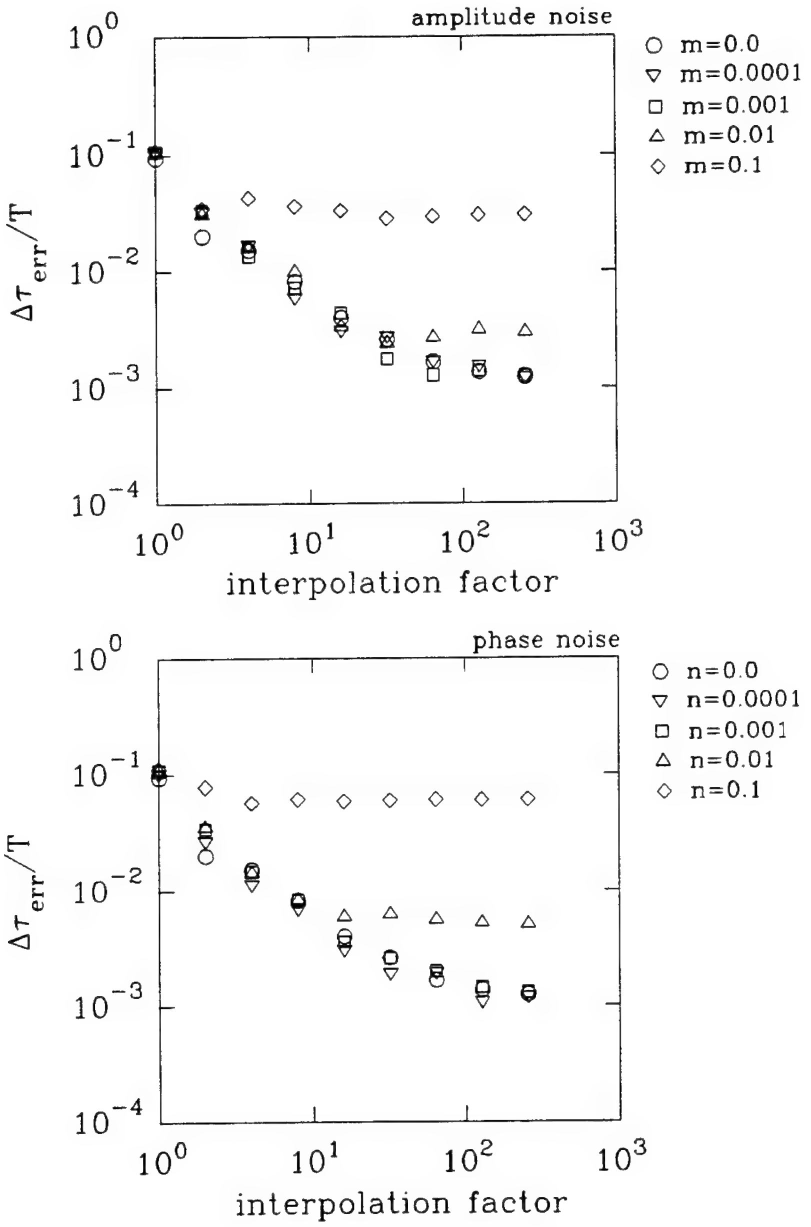
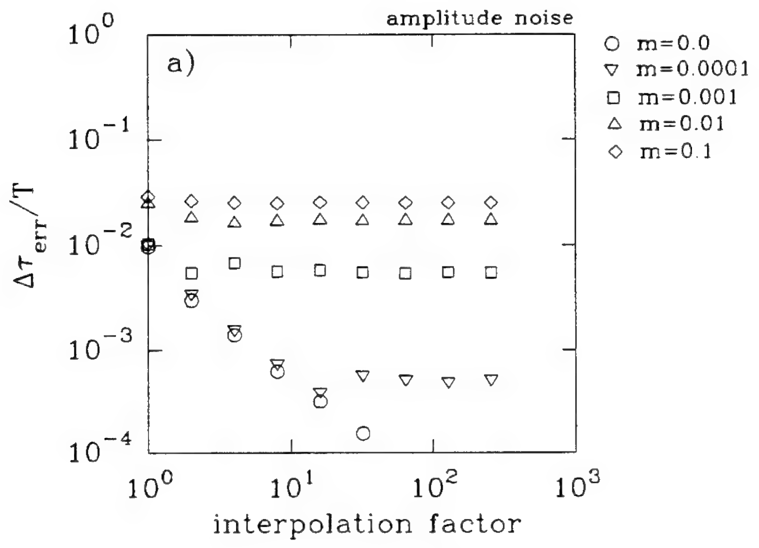


Figure E.2: Error $\Delta\tau_{err}/T$ in the resolution of the location of the peak of the cross-correlation function versus the interpolation factor. T is the acoustic wave period. The number of samples per period is 10 and the number of points used in the spline is 5. a) effect of amplitude noise only, b) effect of phase noise only. The amplitude noise m and the phase noise n are shown on the graph

Figure E.2 showed the effect of noise for a number of points per period of 10 and a number of points in the spline of 5. Figure E.3 shows similar results for a number of points per period of 100. The graphs are qualitatively similar but the effect of noise is more important at the higher sampling rate. If only 5 points are selected in the spline, these 5 points will be located closer to the peak if the sampling rate is higher. This is advantageous in a situation when there is no noise but it is detrimental when noise is present and this is why $\Delta\tau_{err}$ is more sensitive to noise at higher sampling rates.



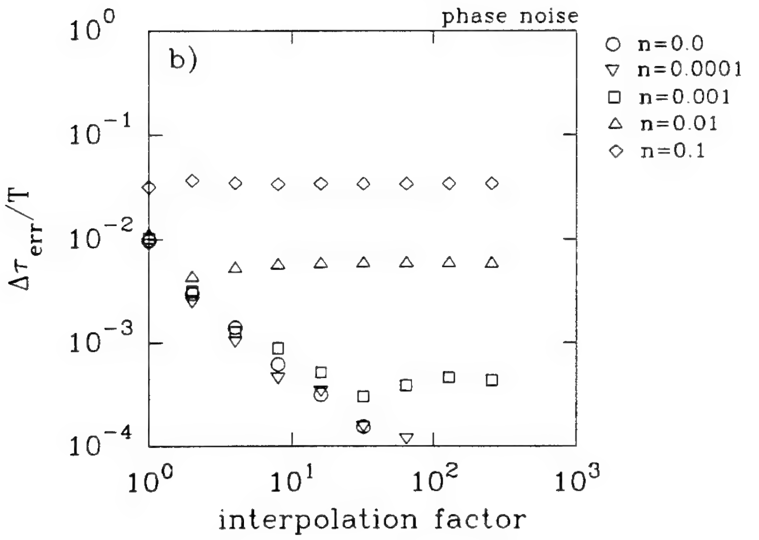


Figure E.3: Error $\Delta\tau_{err}/T$ in the resolution of the location of the peak of the cross-correlation function versus the interpolation factor. T is the acoustic wave period. The number of samples per period is 100 and the number of points used in the spline is 5. a) effect of amplitude noise only, b) effect of phase noise only. The amplitude noise m and the phase noise n are shown on the graph

E.2 Compensating lower sampling rate with an increase in interpolation

In the previous section we have shown that for a same noise level, the interpolation can be pushed further if the data is sampled at a lower rate. This essentially means that a decrease in sampling rate can, to a certain extent, be compensated by an increase in interpolation and hence the same resolution on the time delay can be achieved. We now demonstrate that this can indeed be achieved with the field data.

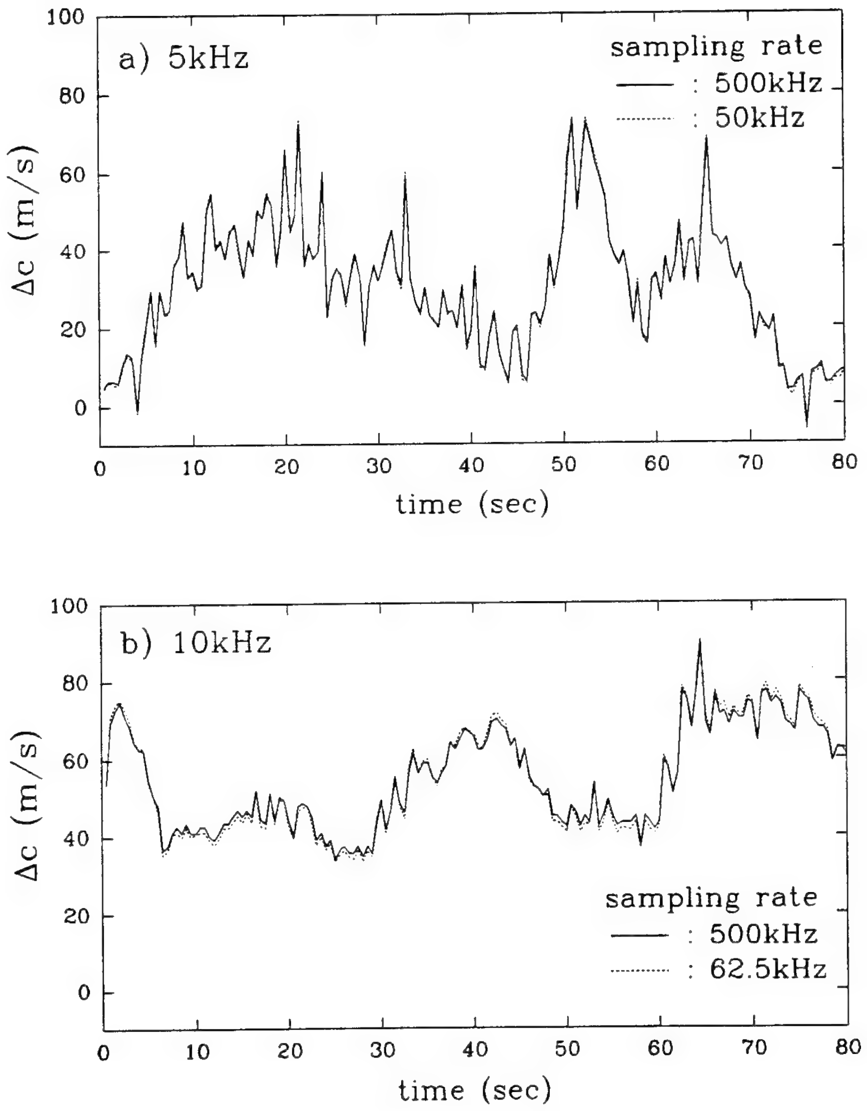


Figure E.4: a) An 80sec time-series of ΔC for 5kHz acoustic pulses sampled at 500kHz (solid line) and 50kHz (dashed line). The decrease in resolution caused by the decrease in sampling rate is compensated by an increase in the interpolation. b) same as in a) but for 10kHz acoustic pulses and for sampling rates which differ by a factor of 8. Note that a) and b) are different time-series.

Figure E.4 shows short time-series of ΔC obtained by sampling the acoustic pulses at 500kHz (solid lines). The dashed lines are obtained by decimating the original data in order to obtain a lower sampling rate and by increasing the interpolation in order to compensate for the lost in resolution caused by the lower sampling rate. The time-series are surprinsingly similar. Note that for lower sampling rates than the ones shown on figure E.4 (i.e. 50kHz and 62.5kHz), the technique fails completely. In general, it was found that the lowest sampling rate should be no less than ~5 times the Nyquist in order for the interpolation to recover the resolution of the higher sampling rate. Of course, this rule of thumb will be a function of the signal to noise ratio. For the data shown in figure E.4, SNR≅100.

Appendix F

Preliminary system for the measurement of sound-speed: Seattle experiment

In this appendix, we briefly report on an experiment conducted in the Pacific (from Seattle) on the measurement of sound-speed. The objective of this field experiment was to test a preliminary system for measuring the speed of sound near the ocean surface. The results from this sea trail pointed out some shortcomings in the system and it suggested many valuable improvements for the design that followed (refer to §6).

Unfortunately, important acoustic interferences, caused mainly by the buoy, severly limited the value of the data in providing a physical understanding of the sound-speed signal near the ocean surface. Nevertheless, it was felt important to briefly review the experiment and its main conclusions.

F.1 The experiment

The experiment was conducted in November 1991 at approximately 100nmi offshore Vancouver Island on the R/V Thompson of the University of Washington in Seattle.

The tethered spar buoy deployed for the measurements of sound-speed is shown in figure F.1. The buoy was built around a 6m long aluminum triangular truss with a 15cm face. A powerful flexural disc transducer model ITC-4004A was mounted at the bottom of the spar. This omnidirectional source was operated at its 2.5kHz resonance frequency. The ITC-1089E receivers (7) were mounted inside the triangular section of the truss in order to protect them from being damaged during deployement operations. A basic measurement consisted of transmitting a short 2.5kHz pulse and to receive it with a vertical array of 7 hydrophones. The time-delay between the received signals at two different hydrophones separated by a fixed and known distance yielded the propagation speed. The time-delay was computed by cross-correlating the two receive signals or by independently cross-correlating the transmit signal with the two receive signals.

Other instrumentation on the spar buoy included an NEC TI-23A video camera mounted inside an underwater Videa Vault (Spring, Texas) enclosure. The video was used to monitor breaking waves at the buoy. A Sea-Bird SBE 3 temperature sensor was mounted on the buoy very close to the water surface. A 150m cable made of multiple shielded pairs was used to carry the hydrophone and temperature signals from the buoy to the ship. The sound source had its own 150m cable in order to prevent cross-talk with the receivers. An underwater enclosure was used to house the hydrophone pre-amplifiers and to serve as a junction box between the receivers and the 150m cable. The pre-amplifiers were based on an AD521J Instrumentation amplifier from Analog Devices.

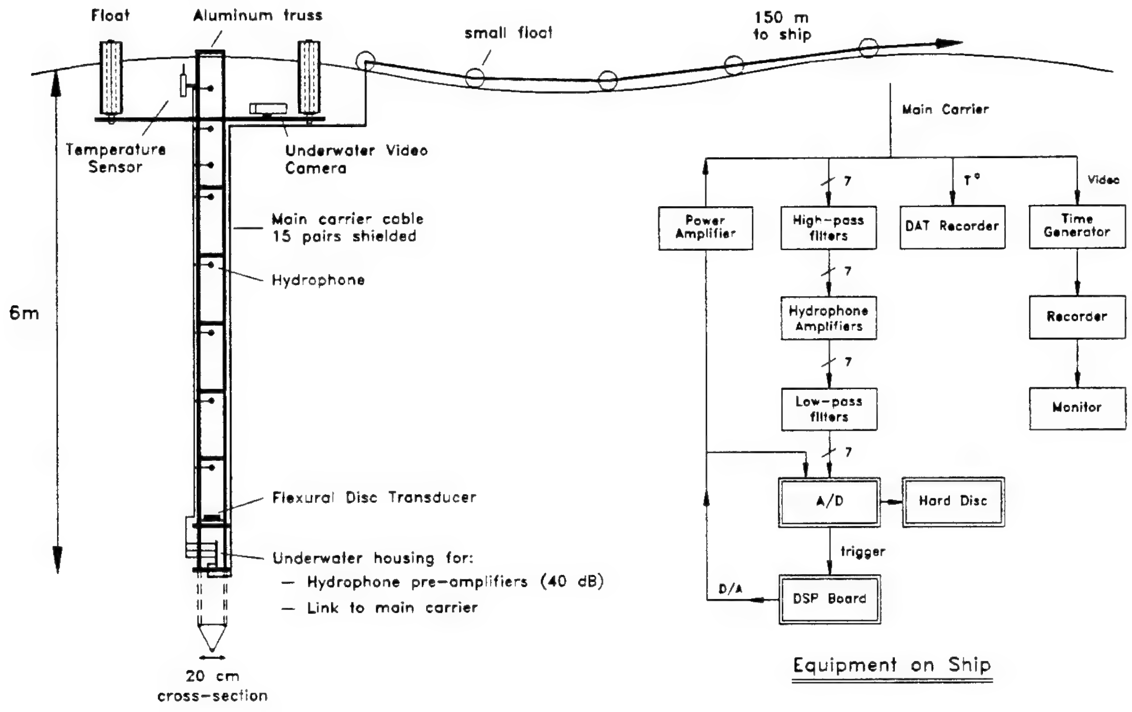


Figure F.1: Sketch of the buoy and instrumentation for the Seattle experiment.

The acoustic pulses were transmitted by a dedicated Spectrum TMS320C30 DSP board. The 2.5kHz pulses were windowed and band-pass filtered before being amplified by a RAMSA WP-9110 power amplifier. The receive signals were pre-amplified by 40dB at the buoy. At the ship, the signals were high-pass filtered at 0.5kHz, amplified by 30dB with low-noise Wicoxon amplifiers and finally low-pass filtered at 5kHz. The transmit and receive signals from 7 hydrophones were sampled at 125kHz/ch and the data were stored to PC RAM. The data in PC RAM were transferred to optical disk every 30sec by a SONY SMO-E501 optical drive. The cross-correlations necessary to extract the time-delay information from the raw signals were performed after the field experiment was completed.

It should be pointed out that the original idea behind the choice of a slender spar type buoy was to reduce acoustic interference by the frame by minimizing the ratio of the frame's cross-sectional dimension to acoustic wavelength. For the present design, this ratio is 1/4. As it will be shown next, this line of thought was not very successful since much acoustic interference was generated by the frame.

F.2 Time-series of acoustic pulses and interference

Figure F.2 shows typical time-series of the transmit and received acoustic pulse as it propagates from the source toward the surface. The hydrophone closest to the source (figure F.2g) receives the acoustic pulse first. The pulse itself is longer than the original transmit signal. This is caused by "ringing" of the transmitter when excited by a short relatively broad-band signal. Nevertheless, the received signal in figure F.2g showed no significant acoustic interference when we compared it to signals transmitted in a laboratory tank. As the pulse propagates upward, it picks up significant acoustic distortions which clearly show up on the time-series. These distortions are believed to be caused by the frame holding the hydrophones. The "acoustic leakage" which smears out the beginning of some of the acoustic pulses may be due to coupling of the acoustic waves with the structure itself.

Post-experiment laboratory tests were conducted to investigate the interference caused by the frame. Half of the spar buoy was immersed in a laboratory tank. Two omnidirectional hydrophones (i.e one transmitter and one receiver) were positioned inside the triangular section of the frame at a distance of 0.4m. The hydrophones were not fixed to the frame; they were suspended so that their heads were positioned in the central axis of the frame. Short pulses similar to the ones described in §6 were transmitted and received. The 5kHz pulse (compared to 2.5kHz for the flexural disk source used in the field) were short enough to eliminate surface reflections from the direct path signal. The results showed that the frame caused significant interference when compared to the results with no frame. The field time-series of the acoustic pulses and the laboratory tests lead to conclude that significant efforts were needed on redesigning the system to minimize acoustic interference.

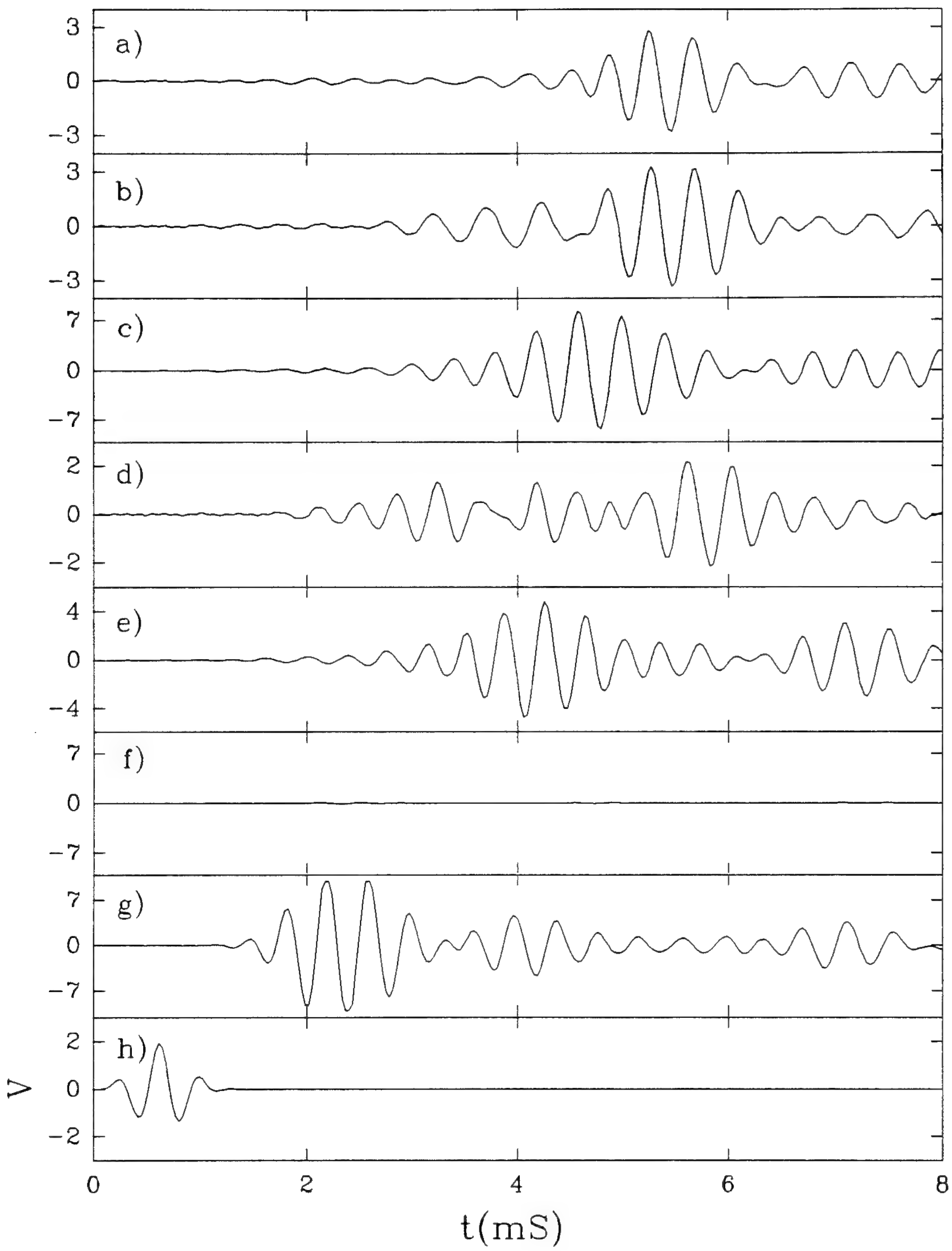


Figure F.2: Time-series of the transmit (h) and receive acoustic pulses. The transducer is located at 6.216m from the surface. The receivers are located at a)0.508m, b)1.016m c)1.524m d)2.032m e)2.540m f)3.530m and g)5.054m from the surface. The hydrophone closest to the transducer receives a clean pulse that shows little signs of acoustic interference. As the pulse moves upward, it picks up significant distortions. The hydrophone in f) was not working.

F.3 Time-delays

Time-delays between the transmit signal and each of the receive signals were computed by finding the peaks of the respective cross-correlation functions. Figure F.3 shows five 10min time-series of the time-delays for wind conditions between 11 to 13m/s. As expected, the average time-delay increases as distance from the source increases but not always in the same proportions. This would indicate that the pulses at certain hydrophones suffer more severely from acoustic interference. The signal fluctuations increase with distance from the source (except for figure F.3d) which would be consistent with an increase in bubble density near the surface. However, the signals should only show fluctuations toward greater time-delays since the effect of bubbles at low frequencies is always to diminish the speed of sound. This is clearly not the case except maybe for figure F.3a which does seem to show fluctuations that increase from their reference (bubble free) value.

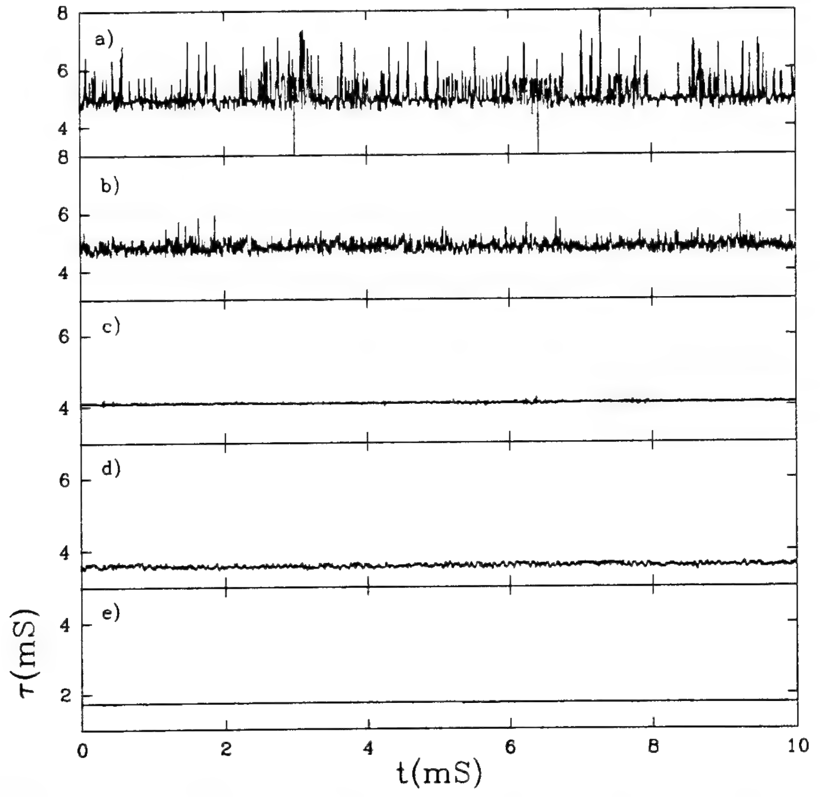


Figure F.3: Time-series of time-delays computed by cross-correlating the transmit signal with the respective received signals. We use the convention that the uppermost receiver is labelled 1 and the lowermost is labelled 7. The time-delays shown are for a) 1, b) 2, c) 3, d) 5 and e) 7. Results for hydrophone 2 and 4 are not shown. The range of the vertical scale is the same for all plots.

These discrepancies suggest that the data were corrupted by acoustic interference from the frame. Several other signal processing schemes such as computing the group velocity instead of the phase velocity did not yield more convincing results.

F.4 Discussion and summary

The results from this preliminary experiment yielded several new ideas for improving the measurement technique. The main point to come out was the importance of rethinking the design to minimize the acoustic interference caused by nearby structural buoy elements.

Although not mentioned in this appendix, the experiment also demonstated the importance of: a) using high quality pre-amplifiers to minimize noise pick up; b) using coaxial cables for carrying the received signals; c) reducing the length of the buoy to facilitate deployments from the ship and d) computing the time-delays in real-time to ensure the recording of the highest quality data and to provide immediate feedback for on-site improvements.

Appendix G
Specifications

This appendix gives detailed specifications on custom made hardware used throughout this study. Furthermore, specifications for highly specialized equipment is also given. Specifications for standard "off-the-shelf" instruments such as digital I/O cards, data acquisition boards and data storage devices can easily be obtained from the manufacturer. Table G.1 gives an overview of the instruments treated in this appendix.

- Danish Hydraulic Institute 80-74G bridge
- Panasonic AG-6300 triggering circuitry
- Superior Electric M061-FD02 stepper motor system
- Sea Bird SBE-3 temperature probe and circuitry
- ITC-1042, ITC-1089, ITC-8181A hydrophones
- Ithaco 144F pre-amplifiers
- RAMSA WP-9110 power amplifier
- Wilcoxon AM-5 low noise amplifier
- Frequency Device 874 high-pass filter
- Frequency Device 844 low-pass filter
- Furuno FCV-561 sonar

Table G.1: Instruments for which specifications appear in this appendix.

Specifications for Danish Hydraulic Institute 80-74G electronic bridge

ouput voltage	±5V asymetrical
output current	±5mA
output impedance	less than 1ohm
temperature range	+10°C to +40°C
output offset	vs temperature: ±1mV/°C vs supply voltage: ±2mV/V
frequency response	0 to 100Hz, 3dB (modified)
gain	1 to 11
zero suppresion	±10V
carrier frequency	3kHz
noise on output	2mV p.p.
warm-up time	10min
linearity	better than 0.5% of full scale

Table G.2: Danish Hydraulic Institute bridge electronics model 80-74G.

Panasonic AG-6300 triggering circuitry

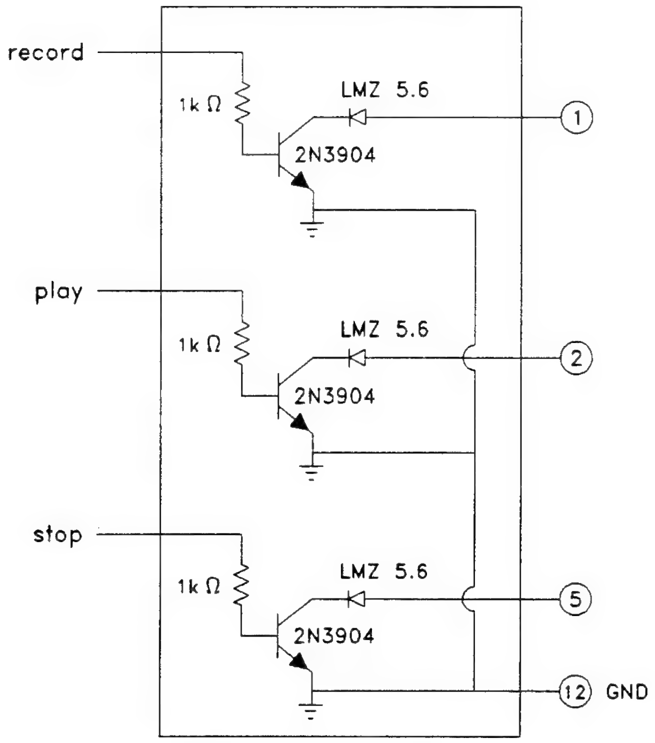


Figure G.1: Circuitry for triggering the Panasonic AG-6300 video recorder. To activate "play" and "stop" ground the appropriate pin. To record, ground both "play" and "record" simultaneously. Pins 1, 2, 5 and 12 are the pin numbers corresponding to the 34pin connector at the back of the recorder. Circuit adapted from Rapp [1986].

Superior Electric M061-FD02 stepper motor and software

A computer controlled stepper motor system was developped to automate the experiments of §3. The entire system was developped by Wesley H. Huang [Huang, 1990]. The main features of the system are reported here for completeness. Details can be found in Huang [1990].

The stepper motor moved a carriage which rolled on rails on the top of the wave channel. The track of the stepper motor was fixed to the side of the channel and a special gear box was built to interface between the stepper motor located on the carriage and the tracks. The stepper motor is from Superior Electric model M061-FD02. It has a resolution of 200 steps per revolution. A SLO-SYN translator module manufactured by Superior Electric model STM-200 was used as the basic interface between the computer and the stepper motor. The role of the translator module is to take a pulse from the computer and make the stepper motor move a single step by energizing its coils. A Metrabyte PIO-24 digital I/O card mounted inside the computer was used to send the appropriate pulse sequences to the translator module. Figure G.2 shows a block diagram of the computerized motor control.

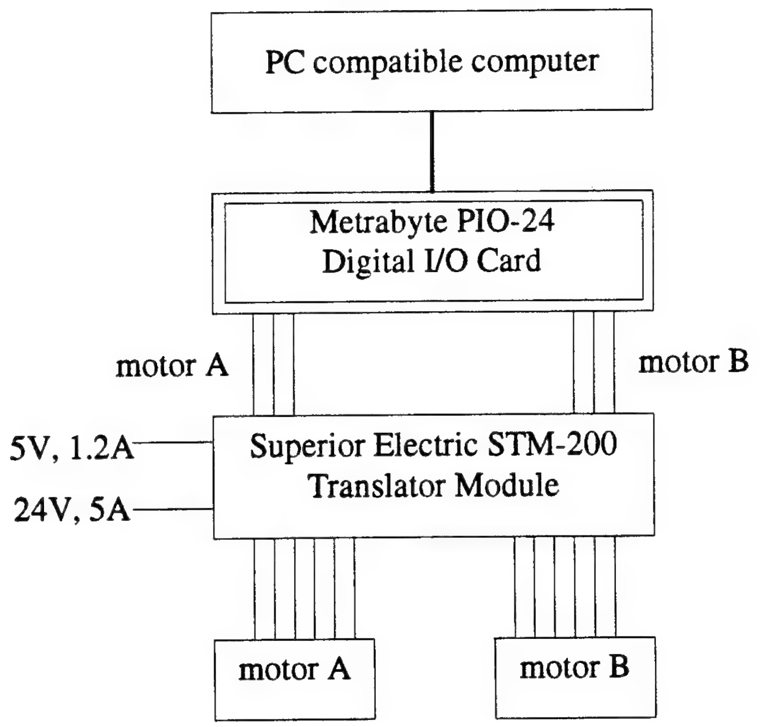

PC compatible computer
Metrabyte PIO-24 Digital I/O Card
motor A
motor B
5V, 1.2A
24V, 5A
Superior Electric STM-200 Translator Module
motor A
motor B

Figure G.2: Block diagram of the computerized motor control. Adapted from Huang [1990].

A software driver written in assembler was used to control the PIO-24 card. The code is listed in appendix H. After calibrating the system, we could obtain an accuracy of ±2mm over a 2m path. The system takes a few minutes to travel a couple of meters. In the context of the laboratory experiment this was not an important issue since several minutes were available between each repeat of the breaking wave experiment.

Sea Bird SBE-3 temperature probe specifications

measurement range	-5.0 to +35.0°C
accuracy/stability	±0.004°C per year (typical) ±0.01°C per 6mo. (guarant.)
resolution	0.0003°C at 24 samples/sec
response time	0.072sec (1.0m/s water vel.) 0.084sec (0.5m/s water vel.)
self-heating	<0.0001°C in still water
warm-up time	<2sec to within 0.005°C
type of output	frequency modulated, 0.7V rms
output frequency range	5kHz to 13kHz

Table G.3: Specifications for Sea Bird SBE-3 temperature sensor.

Sea Bird SBE-3 temperature probe circuitry

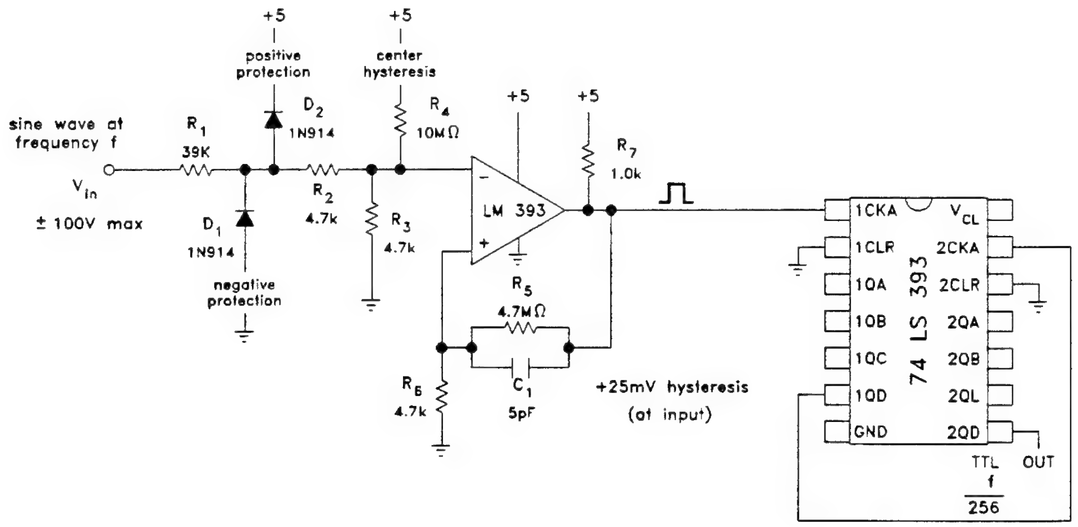
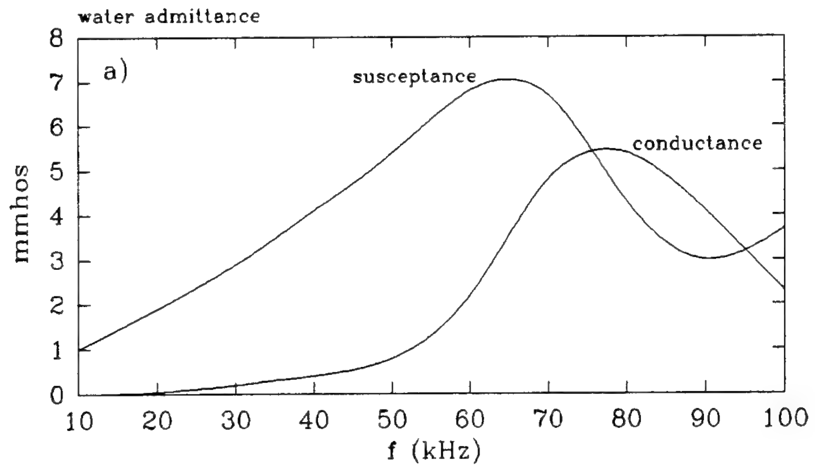


Figure G.1: Circuitry for converting the frequency modulated sine wave output (f) of the SBE-3 temperature probe to a TTL compatible signal of frequency f/256. The TTL signal is used to drive a Metrabyte CTM-PER period counter board which extracts the frequency modulated information contained in the TTL signal. Circuit adapted from Horowitz and Hill [1989, pp242].

ITC-1042 hydrophone

resonance frequency	~80kHz
horizontal beam pattern	omni to ±0.5dB to 80kHz
vertical beam pattern	omni to ±1.5dB to 80kHz
water admittance	see graph below
voltage source level	see graph below
efficiency	50kHz to 100kHz (50% minimum) at resonance
input power	100W around resonance at 10% duty cycle
hydrophone diameter	1.4"
weight in air	~2.2lbs
weigth in water	~1.2lbs

Table G.4: Specifications for ITC-1042 hydrophone



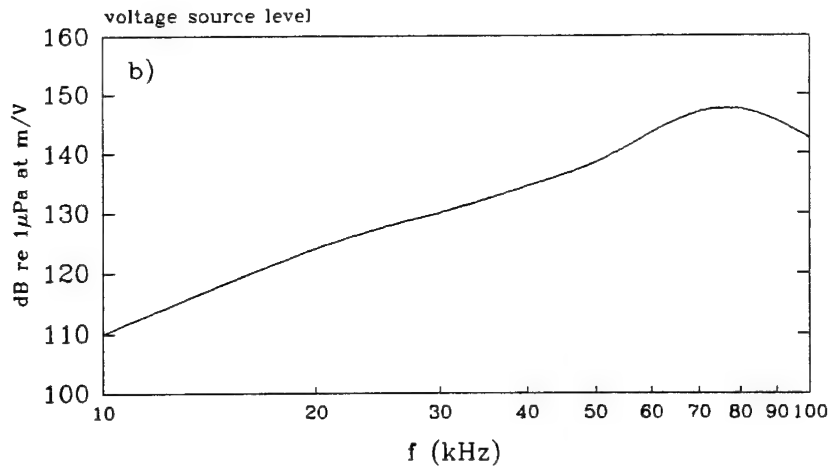


Figure G.2: a) Water admittance of ITC-1042. b) Transmitting Voltage Response of ITC-1042.

ITC-1089E hydrophone

receiving response	±1dB re 1V/μPa to 40kHz (at least)
sensitivity	-212dB re 1V/μPa
horizontal beam pattern	±1dB to 40kHz (at least)
vertical beam pattern	±1dB to 40kHz (at least)
hydrophone diameter	0.5"

Table G.5: Specifications for ITC-1089E hydrophones.

ITC-8181A hydrophone

receiving response	±5dB re 1V/μPa to 40kHz
sensitivity	168dB re 1V/μPa
horizontal beam patternn	omni ±1dB to 20kHz
vertical beam pattern	omni ±3dB to 20kHz
hydrophone diameter	approximately 2"
pre-amplifier gain (unterminated)	23dB
pre-amplifier gain (50ohm load)	18.5dB
preamp noise (100pF source on input) 1Hz bandwidth referred to preamp input	-136dBV at 0.1kHz -155dBV at 1.0kHz -165dBV at 10kHz -167dBV at 100kHz
preamp midband distortion	0.25%
preamp supply	+24V (20 to 28V)
preamp quiescent current	22mA

Table G.6: Specifications for ITC-8181A hydrophone and built-in pre-amplifier.

Ithaco 144F pre-amplifier

gain	-5.0 to +35.0°C
frequency response	±0.004°C per year (typical) ±0.01°C per 6mo. (guarant.)
noise (200kHz bandwidth, referred to input) shorted input 100pF source	0.0003°C at 24 samples/sec
dynamic output impedance	0.072sec (1.0m/s water vel.) 0.084sec (0.5m/s water vel.)
parameter variation with supply voltage	+12V to +25V (max)
supply current (quiescent)	4mA 9mA
max output voltage, Vp-p	8V 18V
max current, Ip	0.75mA 1.5mA
DC output,V	6V 14.0V
distortion (1Vrms in 5Ω)	0.1% 0.1%
power supply isolation	65dB
dimensions	1.13" long by 1.6" diameter
weight	50gm

Table G.7: Specifications for Ithaco 144F pre-amplifier

Ramsa WP-9110 power amplifier

power consumption	~633W at rated output of 150W(rms) + 150W(rms) / 4Ω
rated power output	100W + 100W (8Ω continuous) 150W + 150W (4Ω continuous)
mono-bridge output	300W (8Ω continuous output)
frequency response	20 to 20kHz, ±0.5dB
total harmonic distortion	0.05% or less (8Ω, 100W+100W, 20 to 20kHz) 0.01% or less (4Ω, 150W+150W, 20 to 20kHz) 0.01% or less (bridge, 8Ω, 300W, 20 to 20kHz)
crosstalk	60dB or more (20kHz)
input level	+4dB (continuousy variable) balanced
voltage gain	27dB
input impedance	40kΩ (balanced)

Table G.8: Specifications for Ramsa WP-9110 power amplifier.

Wilcoxon AM-5 low noise amplifier

voltage gain	fixed steps -10, 10, 20, 30, 40, 50, 60dB adjustable -10 to 0dB
Gain accuracy (fixed steps)	±0.2dB
frequency response (-1dB)	1Hz to 80kHz (G=-10 to 30dB) 1Hz to 60kHz (G=40 and 50dB) 1Hz to 40kHz (G=60dB)
amplitidue linearity	<1.0% of full scale
harmonic distortion	<1.0%
input impedance	1000MΩ, 35pF max
input voltage	5V divided by gain setting
output impedance	50Ω
output current	3mA rms max
output noise 1Hz to 50kHz referred to input, battery operated at gains of +30 to +60dB	5μV with 200pF source 3μV with 1000pF source

Table G.9: Specifications for Wilcoxon AM-5 low noise amplifier.

Frequency Devices 874 high-pass filter

response characteristics	Butterworth, 4poles
range (digitally prog)	10Hz to 2.56kHz steps 10Hz
method of programming	8 digital ports
settling time	~10 periods of f_c
input impedance	20kΩ
input voltage range	±10V
output impedance	10Ω
linear operating range	±10V
maximum current	±2mA
offset voltage	2mV typical, 20mV max
noise	50μV
operating voltage	±15V
quiescent current	20mA max

Table G.10: Specifications for Frequency Device model 874 high-pass filter.

Frequency Devices 844 low-pass filter

response characteristics	Butterworth, 4poles
range (digitally prog)	200Hz to 51.2kHz steps of 200Hz
method of programming	8 digital ports
settling time	~10 periods of fc
input impedance	20kΩ
input voltage range	±10V
output impedance	10Ω
linear operating range	±10V
maximum current	±2mA
offset voltage	2mV typical, 20mV max
noise	50μV
operating voltage	±15V
quiescent current	20mA max

Table G.11: Specifications for Frequency Device model 844 low-pass filter.

Furuno FCV-561 sonar

peak power	500W
repetition rate	10Hz
frequency	88kHz
beam width	11°
pulse width	30μsec
minimum range (dead zone)	30 to 40cm
maximum range (adjustable)	10 to 300m

Table G.12: Specifications for the Furuno FVC-561 sonar.

Appendix H
Computer programs used in section I

This appendix contains the computer programs used in section I which includes §1, §3, §4 and §5. All programs were written in Microsoft Fortran V4.1, Microsoft C V6.0 or Microsoft Macro Assembler V5.1. The "make file" containing the compiler and linker directives is given in the comment section at the beginning of each program. Third party library drivers are also described in the comment section. Table H.1 gives an overview of the programs treated in this appendix. A bloc diagram of the programs used in processing the 2D wave channel experiment (§3) is given in figure H.1. The processing for the 3D basin experiment (§4) is similar.

program name	§	description
packet.for	3	Generates the wave packet for the experiment in 2D wave channel.
send20.for	3	Reads the wave packet generated by packet.for and sends the signal to das20 D/A converter to drive the wavemaker.
acqu.for	3	Acquires void-fraction and wave gauge data for the experiment of that chapter 3.
move.for	3	Controls the stepper motors to move the carriage on the tank.
writeout.asm	3	Subroutine of move.for - sends data to Metrabyte PIO-24 card.
expavg.for	3	Makes an ensemble average of three consecutive runs.
newbgmat.for	3	Reconstructs the 2D void-fraction matrix from ensemble averaged time-series.
wgmat.for	3	Reconstructs the wave gauge matrix from ensemble averaged time-series.
grid.for	3	Grids the coarse data contained in *.mat files (generated by newbgmat.for and wgmat.for) onto a finer mesh.
plotexp.for	3	Plots the data generated by grid.
box.for	3	Subroutine of plotexp.for - draws a filled color box
asscol.for	3	Subroutine of plotexp.for - assigns a color based on a threshold of void-fraction
legend.for	3	Subroutine of plotexp.for - plots a legend
axisp.for	3	Subroutine of plotexp.for - plots axis
sumstatb.for	3	Computes moments of the void-fraction field from *.mat files (generated by newbgmat.for and wgmat.for).
statb.for	3	Subroutine of sumstatb.for - computation engine.
swade.for	5	Acquires field void-fraction data for Swade experiment (§5).
com1open.c	5	Function of swade.for - opens com port
tcode.c	5	Function of swade.for - reads Datum time code

Table H.1: Computer programs treated in this appendix.

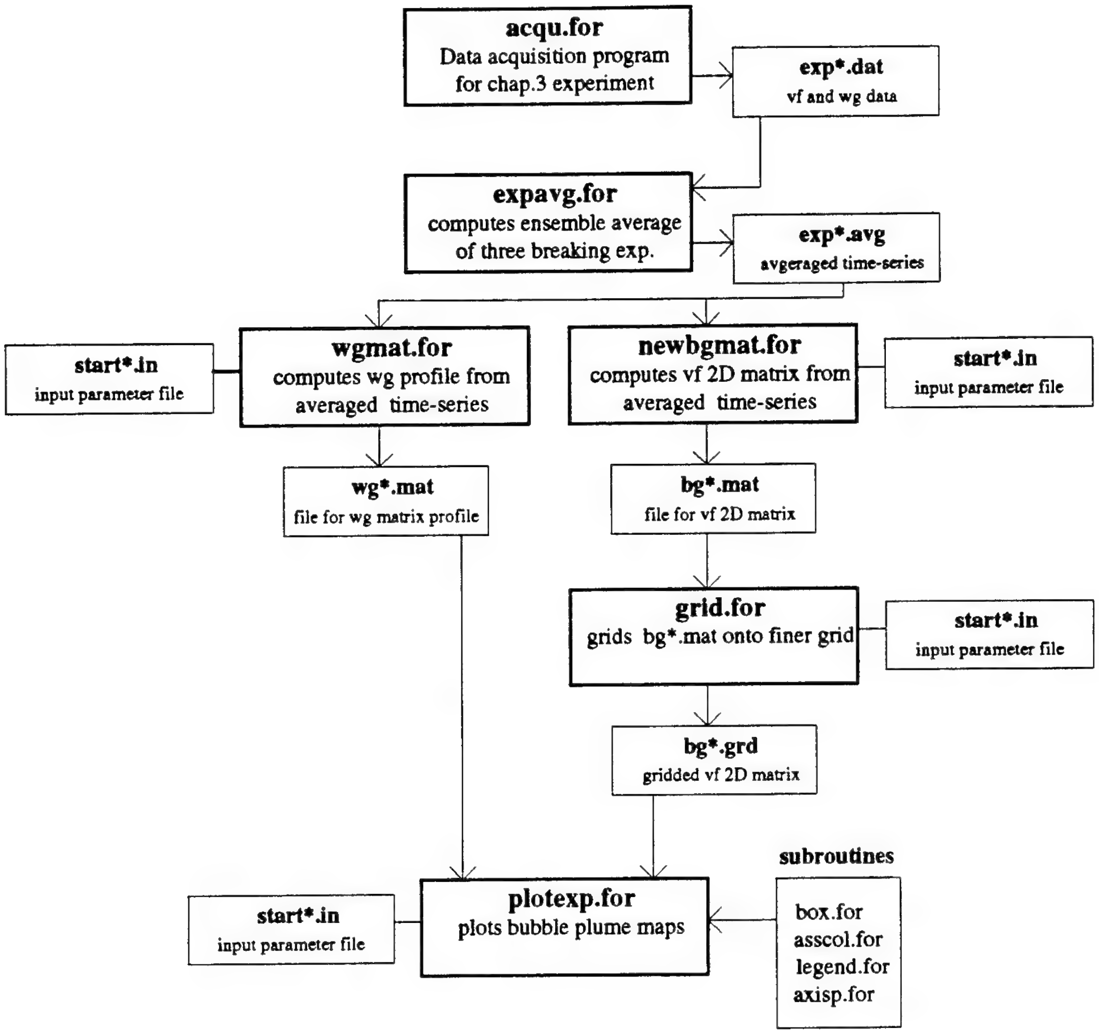


Figure H.1: Bloc diagram of programs organization for processing and plotting the bubble plume data of the 2D experiment (§3). Note that the processing of the data for the 3D experiment (§4) is essentially similar. Bold boxes are programs, all other boxes are input files, output files or subroutines. The (*) refers to various file numbers.

program: packet.c

```
C
C    PROGRAM PACKET
C
C   PROGRAM TO GENERATE A PACKET OF WAVES
C   COMPOSED OF 32 SINUSOIDAL COMPONENTS
C
C   INP:    fc: center frequency of packet (Hz)
C           bw: bandwidth of packet (Hz)
C           xb: breaking position (m)
C           tb: time to breaking (sec)
C           pd: period (sec) of entire signal (pd<25),
C                npt=pd/(1/fout)
C           flag:
C                if yes => slope (ak) is kept constant
C                if no  => amplitude (aa) is kept constant
C           tf:
C                if yes => transfer function is applied
C                     (ampl. & phase)
C                if no  => no transfer function is applied
C   DATA:
C           nfc: number of frequ. compon evenly spaced (32)
C           ak: constant (non dimensional)
C           aa: constant (cm)
C           npt: number of points forming the packet (2000)
C           fout: output frequency of d/a board (100 Hz)
C           h: depth of water (0.6m)
C   OUTPUT: data: array containing packet
C
C   08-25-90: FIRST WRITTEN BY ERIC LAMMARE
C   09-03-90: MODIFIED
C
C   COMPILING AND LINKING NOTES:
C   packet.obj: packet.for
C     FL /c /4Ydb /G2 /4I2 /Fs packet.for
C
C   packet.exe: packet.obj
C     LINK $**, $@, NUL, c:\lib\llibfor7+llibc7 /NOD /NOE
C
      program packet
C
      integer nfc, npt
      integer i, j, n
C
      real aa, ak, fc, bw, xb, tb, g, fout, pd
      real f(32), a(32), pha(32), k(32), corrpha(32), data(16000)
      real deltabw, pi, ko, time, signal, h, w2g
      real a0, a1, a2, a3
      real a0pha, alpha
      real sumak, nakc
C
      character*30 filename
      character*1 flag, tf
C
      data nfc/32/, fout/100.0/, aa/1.0/, ak/0.01/, g/9.81/, h/0.60/
      data pi/3.14159/
      data a0/-2.718/, a1/21.184/, a2/-17.125/, a3/4.055/
      data a0pha/1.14/, alpha/-1.13/
C
C   INPUT DATA
C
      write(*,'(a\)') ' Enter output filename: '
      read (*,'(a)') filename
      write(*,'(a\)') ' Enter fc (Hz): '
      read (*,*) fc
      write(*,'(a\)') ' Enter bw (Hz): '
      read (*,*) bw
      write(*,'(a\)') ' Enter xb (m) : '
      read (*,*) xb
      write(*,'(a\)') ' Enter tb(sec): '
      read (*,*) tb
      write(*,'(a\)') ' Enter pd(sec): '
      read (*,*) pd
      npt = pd / (1/fout)
      write(*,'(a\)') ' Keep ak constant (y/n): '
      read (*,'(a)') flag
      if (flag .eq. 'n' .or. flag .eq. 'N')
     1   print *,'Therefore amplitude is kept constant'
      write(*,'(a\)') ' Apply transfer function (y/n): '
      read (*,'(a)') tf
      print *,'Computing ...'
      if (tf .eq. 'n' .or. tf .eq. 'N') then
        a0 = 1.0
        a1 = 0.0
        a2 = 0.0
        a3 = 0.0
        alpha = 0.0
        print *,'No transfer function applied'
      endif
C
C   COMPUTE FREQUENCY COMPONENTS [EVENLY
SPACED IN FREQ. DOMAIN]
C
      deltabw = bw / (nfc-1)
      do 10 i = 0, nfc - 1
        f(i+1) = (fc - bw/2.0) + deltabw * i
   10 continue
C
C   COMPUTE WAVENUMBER FOR EACH FREQ. COMP.
USING DISPERSION RELAT.
C
      do 20 i = 1, nfc
        w2g = ((2*pi*f(i))**2) / g
        ko = w2g
   25   k(i) = w2g / tanh(ko*h)
        if (abs(k(i)-ko) .lt. 0.0001) go to 20
        ko = k(i)
        go to 25
   20 continue
C
C   COMPUTE PHASE OF EACH COMPONENT WITH
CORRECTION FOR TRANS. FUNCT.
C    a0pha (rad), alpha (rad/Hz)
C
      do 30 i = 1, nfc
        corrpha(i) = a0pha - alpha*f(i)
   30 continue
C
C   IF ak CONSTANT OPTION IS CHOSEN THEN ADJUST
a SUCH THAT ak
C   IS CONSTANT AND THEN CORRECT FOR
AMPLITUDE TRANSFER FUNCTION
```

```
C   IF a CONSTANT OPTION IS CHOSEN THEN SIMPLY
APPLY TRANSF.
C   FUNCT. TO a
C   a0, a1, a2, a3 ARE COEFF. TO THE  AMPL. TRANSF.
FUNCT.
C
    do 40 i = 1, nfc
      if (flag .eq. 'y' .or. flag .eq. 'Y') then
        a(i) = ak / (k(i)/100)
      else
        a(i) = aa
      endif
      a(i) = a(i) / (a0 + a1*f(i) + a2*f(i)**2 + a3*f(i)**3)
  40 continue
    sumak = 0.0
    nakc = 0.0
    if (flag .eq. 'y' .or. flag .eq. 'Y') then
      do 45 i = 1, nfc
        sumak = sumak + a(i) * (a0 + a1*f(i) + a2*f(i)**2 +
   1          a3*f(i)**3) * (k(i)/100)
  45   continue
    else
      do 46 i = 1, nfc
        nakc = nakc + a(i) * (a0 + a1*f(i) + a2*f(i)**2 +
   1          a3*f(i)**3)
  46   continue
      nakc = nakc * (k(16)/100)
    endif
C
C   COMPUTE SIGNAL BY SUPERPOSING ALL NFC
FREQUENCY COMPONENT
C
    do 50 j = 1, npt
      time = (j-1) * (1/fout)
      signal = 0.0
      do 60 i = 1, nfc
        signal = signal + a(i) * cos( - k(i)*xb
   1              - 2*pi*f(i)*(time - tb) - corrpha(i) )
  60   continue
      data(j) = signal
  50 continue
C
C   SET EXPONENTIAL INCREASE AND DECAY USING
1.0 SEC (100 POINTS)
C
    n = 100
    do 55 i = 1, n
      data(i) = data(n) * exp((i - n)/20.0)
  55 continue
    do 57 i = npt-n, npt
      data(i) = data(npt-n) * exp(((npt-n) - i)/20.0)
  57 continue
C
C   WRITE DATA TO DISK
C
    open(16, file=filename, status = 'unknown')
    write(16,4000) 'fc  (Hz)  = ',fc
    write(16,4000) 'bw  (Hz)  = ',bw
    write(16,4000) 'xb  (m)   = ',xb
    write(16,4000) 'tb  (sec) = ',tb
    write(16,3000) 'nfc (#)   = ',nfc
    write(16,4000) 'fout(Hz)  = ',fout
    write(16,3000) 'npt (#)   = ',npt
    write(16,4000) 'h   (m)   = ',h
C
    if (flag .eq. 'y' .or. flag .eq. 'Y') then
      write(16,4000) 'ak (nd)   = ',ak
      write(16,4000) 'sumak (nd)= ',sumak
      write(16,5000) 'const. ak  '
      write(16,5000) 't.f. applied'
    else
      write(16,4000) 'aa  (cm)  = ',aa
      write(16,4000) 'Nakc (nd) = ',nakc
      write(16,5000) 'const. amp  '
      write(16,5000) 'no t.f.    '
    endif
    write(16,5000) '            '
    do 70 j = 1, npt
      write(16,2000) data(j)
  70 continue
1000 format(i10)
2000 format(f10.3)
3000 format(a15,i10)
4000 format(a15,f10.3)
5000 format(a15)
C
    stop
    end
```

program: send20.for

```
C
C   PROGRAM SEND20
C
C   PROGRAM TO SEND TO THE WAVEMAKER
C   A PACKET OF WAVES
C   GENERATED BY THE PROGRAM PACKET.FOR
C   THIS PROGRAM IS WRITTEN FOR DAS20 BOARD
C
C   NOTE THAT THE D/A OUTPUT ON BOARD MUST BE
C   SET TO +/- 10 VOLTS
C
C   INPUT:
C      filename: packet filename (generated by packet.for)
C      gain: gain factor to be applied to signal
C      dout0: time for digital output (100th sec)
C   OUTPUT:
C      to channel 0 for d/a and to the specified channel
C      for dout?
C
C   Special lib: das20.lib by Metrabyte for their das20 board
C
C   08-21-90: FIRST WRITTEN BY ERIC LAMMARE
C   09-03-90: MODIFIED
C   09-09-90: MODIFIED FOR FALL 90 EXP.
C
C   COMPILING AND LINKING NOTES:
C   send20.obj: send20.for
C    FL /c /4Ydb /G0 /4I2 /Fs send20.for
C
C   send20.exe: send20.obj
C    LINK $**, $@, NUL, c:\lib\llibfor7+llibc7+das20f /NOD
C    /NOE
C
    program send20
C
    integer base
    integer*4 buffer, alloc
    integer choice
```

```
      integer das20
      integer data(0:2499)
      integer dmalev
      integer dout0
      integer i
      integer intlev
      integer junkint
      integer*4 k
      integer mode0, mode7, mode8, mode9, mode12, mode15,
mode25
      integer n1rate, n2rate
      integer npt
      integer param(5)
      integer rcode
      integer rcyc
      integer segptr, offadr
C
      real datar(0:2499)
      real gain
      real junkreal
      real sumak
C
      character*30 filename
      character*15 junkname
C
      data base/#300/
      data dmalev/1/
      data dout0/1/
      data intlev/3/
      data mode0/0/, mode7/7/, mode8/8/, mode9/9/
      data mode12/12/, mode15/15/, mode25/25/
      data n2rate/1000/, n1rate/50/
      data rcyc/1/
      buffer = alloc(32766)
C
C   INTIALIZE DAS16 A/D BOARD
C
      param(1) = base
      param(2) = intlev
      param(3) = dmalev
      rcode = das20(mode0, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C   SET ANALOG CHANNEL 0 TO 0.0 VOLTS
C
C   DIGITAL PORT 0: LOW  (NO TRIGGERING OF DAS16
D/A)
C   DIGITAL PORT 1: HIGH (PLAY/REC OF VIDEO
ACTIVATED)
C   DIGITAL PORT 2: LOW  (STOP OF VIDEO NOT
ACTIVATED)
C   DIGITAL PORT 3: HIGH (TIMECODE SET TO PAUSE)
C   THEREFORE THE DIGITAL NUMBER BECOMES (2+8)
= 10
C   ALSO, DUMMY LOOP TO GIVE TIME TO VIDEO TO
START
C
      param(1) = 0
      param(2) = 0
      rcode = das20(mode7, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
      param(1) = 10
      rcode = das20(mode15, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C   INPUT WAVE FILE
C
      write(*,'(a\)') ' Enter wave data file: '
      read (*,'(a)') filename
      write(*,'(a\)') ' Enter Gain: '
      read (*,*) gain
C
C   LOAD DATA AND ADJUST FOR A/D AND ADJUST
FOR GAIN
C
      open(16, file = filename, status = 'old')
      read(16,4000) junkname, junkreal
      read(16,4000) junkname, junkreal
      read(16,4000) junkname, junkreal
      read(16,4000) junkname, junkreal
      read(16,3000) junkname, junkint
      read(16,4000) junkname, junkreal
      read(16,3000) junkname, npt
      read(16,4000) junkname, junkreal
      read(16,4000) junkname, junkreal
      read(16,4000) junkname, sumak
      write(*,6000) ' Slope of packet = ', gain * sumak
      read(16,4000) junkname
      read(16,4000) junkname
      read(16,4000) junkname
      do 10 i = 0, npt-1
        read(16,2000) datar(i)
  10  continue
 1000 format(i10)
 2000 format(f10.3)
 3000 format(a15,i10)
 4000 format(a15,f10.3)
 5000 format(a15)
 6000 format(a19,f6.3)
C
 102  continue
      if (gain .gt. 4.0) then
        print *, 'This gain is too high'
        stop
      endif
      do 15 i = 0, npt-1
        data(i) = int(gain * datar(i) * (4095/20.0))
  15  continue
C
C   SET CLOCK RATE FOR DAS20 A/D BOARD
C
      param(1) = n1rate
      param(2) = n2rate
      rcode = das20(mode25, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C   LOAD DATA WITH MODE8 INTO buffer MEMORY
SEGMENT
C
 101  continue
      param(1) = npt
      param(2) = segptr(buffer)
      param(3) = 0
      param(4) = offadr(data(0))
      rcode = das20(mode8, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C   SEND DATA TO D/A CHANNEL 0
C
      param(1) = npt
      param(2) = segptr(buffer)
      param(3) = 2
      param(4) = rcyc
      param(5) = 0
      pause 'Press return to send signal now'
      print *,'Signal being send'
      print *
      rcode = das20(mode9, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C    MONITOR DATA OUTPUT COUNT NUMBER AND
OUTPUT DIGITAL DATA
C    SET DIGITAL PORTS
C    DIGITAL PORT 0: HIGH (TRIGGERING OF DAS16 D/A)
C    DIGITAL PORT 1: HIGH (PLAY/REC OF VIDEO
ACTIVATED)
C    DIGITAL PORT 2: LOW  (STOP OF VIDEO NOT
ACTIVATED)
C    DIGITAL PORT 3: LOW  (TIMECODE SET TO ON)
C    THEREFORE THE DIGITAL NUMBER BECOMES (1+2)
= 3
C
  104 continue
      rcode = das20(mode12, param)
      if (param(3) .eq. dout0) then
        param(1) = 3
        rcode = das20(mode15, param)
        go to 103
      endif
      go to 104
  103 continue
      rcode = das20(mode12, param)
      if (param(3) .lt. npt) go to 103
C
C    WRITE TO DIGITAL PORTS
C    DIGITAL PORT 0: LOW  (NO TRIGGERING OF DAS16
D/A)
C    DIGITAL PORT 1: LOW  (PLAY/REC OF VIDEO
ACTIVATED)
C    DIGITAL PORT 2: HIGH (STOP OF VIDEO NOT
ACTIVATED)
C    DIGITAL PORT 3: HIGH (TIMECODE SET TO ON)
C    THEREFORE THE DIGITAL NUMBER BECOMES (4+8)
= 12
C    DUMMY LOOP TO GIVE A FEW MORE SECONDS OF
RECORDING
C
      do 30 k = 1, 150000
   30 continue
      param(1) = 12
      rcode = das20(mode15, param)
      if (rcode .ne. 0) print *,'Error, rcode = ', rcode
C
C    OPTION TO GENERATE 1) SAME WAVE, 2) CHANGE
GAIN, 3) EXIT
C
      print *, '1) Generate same wave'
      print *, '2) Change gain and generate same wave'
      print *, '3) Exit program'
      write(*,'(a\)') ' Enter choice: '
      read (*,*) choice
      if (choice .eq. 1) go to 101
      if (choice .eq. 2) then
        write(*,'(a\)') ' Enter new Gain: '
        read (*,*) gain
        write(*,6000) ' Slope of packet = ', gain * sumak
        go to 102
      endif
C
      if (choice .eq. 3) stop
C
      stop
      end
```

program: acqu.for

```
C
C    PROGRAM ACQU
C
C    PROGRAM TO ACQUIRE DATA FOR BG EXP.
C    OF FALL 1990
C
C    CHANNEL 0,1,2: WG DATA (DMA CHANNELS)
C    CHANNEL 3,4,5: BG DATA (DMA CHANNELS)
C    CHANNEL 6,7: AIR AND WATER TEMPERATURE
C    (SAMPLED AT THE END)
C
C    NOTES:
C        1) BOARD MUST BE SET IN BIPOLAR MODE
C        2) NUMBER OF DATA SAMPLED IN EACH DMA
C             CHAN = 8000
C        3) SAMPLING RATE = 200 Hz
C        4) DMA DATA STARTS AT DATA(101)
C        5) DATA(1) TO DATA(100) USED FOR OTHER
C             PARAMETERS
C        6) SAMPLING START WHEN DIGITAL INPUT
C             PORT 0 GOES HIGH
C    Example of batch file using Key-fake program
C        key-fake "850" 13 "n" 13
C        acqu
C
C    Special lib: das16f.lib by Metrabyte for their das16 board
C
C    09-03-90: FIRST WRITTEN BY ERIC LAMARRE
C    09-15-90: MOD. TO MINIMIZE BATCH FILE INPUT
C
C    COMPILING AND LINKING NOTES
C    acqu.obj: acqu.for
C     FL /c /4Ydb /G2 /Fs acqu.for
C    acqu.exe: acqu.obj
C     LINK $**, $@, NUL, c:\lib\llibfor7+llibc7+das16f /NOD
C     /NOE;

C
      program acqu
C
      integer baddr
      integer base
      integer buf, halfbuf
      integer*2 buffer(32768)
      integer*2 das16
      integer*2 data(48100)
      integer dmalev
      integer fch
      integer i, j, k, l
      integer*2 idecode
      integer intlev
      integer*2 mode0, mode1, mode3, mode6, mode7, mode8,
mode17
```

```
      integer*2 mode13
      integer nch
      integer ncyc
      integer n1rate, n2rate
      integer*2 param(5)
      integer*2 rcode
      integer rcyc
      integer samples
      integer*2 segadr
      integer sch
      integer trig
C
      real clock
      real srate, tsrate
      real sumat, sumwt
C
      character*1 answer
      character*30 filename
C
      data base/#340/
      data clock/1000000.0/
      data dmalev/1/
      data intlev/3/
      data mode0/0/, mode1/1/, mode3/3/, mode6/6/, mode7/7/
      data mode8/8/, mode13/13/, mode17/17/
      data n1rate/2/
      data trig/1/
      data rcyc/1/
C
      data srate/200.0/
      data sch/0/, fch/5/
      data samples/8000/
      data buf/100/
C
C   INPUT OF PARAMETERS + COMPUTATION OF
OTHER PARAMETERS
C
      call getdat(data(2),data(3),data(4))
      open(16, file = 'info.cal', status = 'old')
      read(16,*) data(18), data(19)
      read(16,*) data(20), data(21)
      read(16,*) data(22), data(23)
      read(16,*) data(24), data(25), data(26), data(27)
      read(16,*) data(28), data(29), data(30), data(31)
      read(16,*) data(32), data(33), data(34), data(35)
      read(16,*) data(47), data(48), data(49),
     1        data(50), data(51), data(52)
      close(16)
C
      write(*,'(a\)') ' Enter bg2 position: '
      read (*,*) data(38)
      open(16, file='superfil.fil', status = 'old')
      read(16,*) data(1)
      read(16,*) data(37)
      read(16,*) data(39)
      read(16,*) data(41)
      close(16)
      data(36) = data(38) - 20
      data(40) = data(38) + 20
      data(42) = 300
      data(43) = data(36) - 40
      data(44) = data(40) + 40
C
      nch = fch - sch + 1
      halfbuf = buf / 2
      tsrate = nch * srate
      n2rate = nint(clock / (n1rate*tsrate))
      srate = (clock / (n1rate * n2rate)) / nch
      ncyc = nch*samples / buf
C
C   INITIALIZE DAS-16 A/D BOARD
C
      param(1) = base
      param(2) = intlev
      param(3) = dmalev
      rcode = das16(mode0, param)
      if(rcode .ne. 0) print *,' Error mode0, rcode = ',rcode
C
C   SET CHANNEL SCAN LIMITS
C
      param(1) = sch
      param(2) = fch
      rcode = das16(mode1, param)
      if(rcode .ne. 0) print *,' Error mode1, rcode = ',rcode
C
C   SET CLOCK RATE
C
      param(1) = n1rate
      param(2) = n2rate
      rcode = das16(mode17, param)
      if(rcode .ne. 0) print *,' Error mode17, rcode = ',rcode
C
C   SET UP BUFFER POINTER
C
      baddr = segadr(buffer)
C
C   BEGIN DMA CONVERSION TO buffer
C
      param(1) = buf
      param(2) = baddr
      param(3) = trig
      param(4) = rcyc
      print *,'Waiting for trigger on IP0 ... '
      rcode = das16(mode6, param)
      print *,'Sampling ... '
      call gettim(data(5), data(6), data(7), data(8))
      if(rcode .ne. 0) print *,' Error mode6, rcode = ',rcode
C
C   CHECK IF HALF BUFFER FILLED, IF YES INITIATE
TRANSFER
C   FROM buffer TO data ARRAYS
C
      do 10 i = 1, ncyc
        do 10 j = 1, 2
 100      rcode = das16(mode8, param)
C         do 30 l = 1, 10
C 30      continue
          if (j .eq. 1 .and. param(3) .le. halfbuf .or.
     1       j .eq. 2 .and. param(3) .gt. halfbuf) then
            go to 100
          else
C             TRIGGER COMPUTER THAT ACQUIRES
ACOUSTIC DATA
            if (i .eq. 167 .and. j .eq. 2) then
              param(1) = 1
              rcode = das16(mode13, param)
            endif
C
            if (i .eq. ncyc .and. j .eq. 2)
     1        call gettim(data(9), data(10), data(11), data(12))
```

```
          do 20 k = 1, halfbuf
            data(halfbuf * (2*(i-1)+(j-1)) + k + 100) =
     1                idecode(buffer(halfbuf*(j-1)+k))
  20      continue
        endif
 10 continue
C
C   SHUT DOWN DMA
C
      rcode = das16(mode7, param)
      if(rcode .ne. 0) print *,' Error mode7, rcode = ',rcode
      write(*,2000) data(5), data(6), data(7), data(8)
      write(*,2000) data(9), data(10), data(11), data(12)
 2000 format(' ',i2.2,':',i2.2,':',i2.2,':',i2.2)
C
C   SAMPLING OF AIR AND WATER TEMPERATURE
C   MUST SET A NEW CHANNEL SCAN LIMIT
C
      param(1) = 6
      param(2) = 7
      rcode = das16(mode1, param)
      if(rcode .ne. 0) print *,' Error mode1, rcode = ',rcode
C
      sumat = 0.0
      sumwt = 0.0
      do 40 i = 1, 30000
        rcode = das16(mode3, param)
        if(rcode .ne. 0) print *,' Error mode3, rcode = ',rcode
        sumat = sumat + param(1)
        rcode = das16(mode3, param)
        if(rcode .ne. 0) print *,' Error mode3, rcode = ',rcode
        sumwt = sumwt + param(1)
  40  continue
      data(45) = int(sumat / 30000)
      data(46) = int(sumwt / 30000)
C
C   WRITING THE DATA TO DISK + OPTION TO MODIFY
WG POSITION
C
      write(*,'(a\)') ' Want to input wg position (y/n): '
      read (*,'(a)') answer
      if (answer .eq. 'y' .or. answer .eq. 'Y') then
      do 50 i = 1, 3
        print *, ' Enter position of wg',i
        read (*,*) data(41+i)
  50  continue
      endif
C
      open(16, file='superfil.fil', status = 'old')
      write(16,*) data(1)+1
      write(16,*) data(37)
      write(16,*) data(39)
      write(16,*) data(41)
      close(16)
C
      write(filename, 1000) data(1)
 1000 format('c:\exp\exp', i5.5)
      open(16,file=filename // '.dat',form='binary',access='direct',
     1      recl=1924, status = 'new')
      write(16) data
      close(16)
C
      open(16,file='test',status='unknown')
      do 300 i = 1, 48100
        write(16,7000) data(i)
 300  continue
 7000 format(i8)
      close(16)
C
      stop
      end
C
C   FUNCTION IDECODE
C
C   CONVERTS 12 BITS DATA INTO 16 BITS DATA
C
      integer*2 function idecode(ibin)
        integer*2 ibin
        idecode = isha(ibin, -4) - sign(2048, ibin)
      end
```

program: move.for

```
C Program move.for
C Program to control the stepper motor which
C moves the carriage on top of the wave channel
C
C Uses and assembly routine writeout.asm
C to send instructions to the Metrabyte PIO-24 card.
C
C Compiler and linker directives:
C move.obj: move.for
C         FL /c /4Ydb /G2 /Fs move.for
C
C move.exe: move.obj pulse.obj writeout.obj
C     LINK $CC , $@, NUL, c:\lib\llibfor7 /NOD /NOE;
C
C writeout.obj: writeout.asm
C   masm writeout ;
C
      interface to subroutine writeout(addr,byt)
      integer*2 addr,byt
      end
C
      program move
C
      integer*2 a,b
      integer pulses, n
      integer dir
      integer delay, coast
      integer width
      integer mask, off
      integer which
      integer base
      integer k,l
C
      data width/4/, base/864/
C
C   initialize the output portA to be all outputs
C
      a = base + 3
      b = 128
      call writeout(a,b)
C
C   initialize the motor to be off
C
      a = base
      b = 1 + 8
      call writeout(a,b)
```

```
C
C   enter channel to be used and
C   enter number of pulses to be sent (80 pulses = 1 cm)
C
    print *, 'Enter channel (1 or 2)'
    read(*,*) which
    print *, 'Enter number of pulses (positive=right,
negative=left)'
    read(*,*) pulses
    print *, 'Enter coasting delay (i.e. 1000)'
    read(*,*) coast
C
C   find direction from sign of pulses
C
    if (pulses .lt. 0) then
C     go left
    dir = 2
    else
C     go right
    dir = 4
    endif
C
C   assign channel to be operated
C
    if (which .eq. 1) then
C use motor 1
    mask = dir + 8
    off = 1
    else
C use motor 2
      mask = dir*8 + 1
    off = 8
    endif
C
C   move cart
C
    n = abs(pulses)
    a = base
    do 10 l=1,n
    b= mask
    call writeout(a,b)
    do 20 k=1,width
  20  continue
    b= mask + off
    call writeout(a,b)
C     exponential acceleration
C     delay = int( 5000*exp(-float(n)/25.0) )
C     if (delay .lt. coast) then
C       delay = coast
C     endif
      delay = coast
    do 30 k=1,delay
  30  continue
  10 continue
C
C   turn motor off
C
    a = base
    b = 9
    call writeout(a,b)

    end
```

subroutine: writeout.asm

```
; this procedure is called by a fortran program.
; writeout(A,B) will place B into
; the memory location A.  Adapted an the example in the mixed
; language reference manual.
;
; Written 6/90 by Wes Huang(1990)
.model large
.code
    public   writeout
writeout     proc
    push     bp
    mov      bp,sp

    les  bx,[bp+10]
    mov      dx,es:[bx]
    les  bx,[bp+6]
    mov      ax,es:[bx]
    out  dx,al

    pop bp
    ret  8
writeout     endp
    end
```

program: expavg.for

```
C
C   PROGRAM expavg
C
C   TAKE 3 TIME-SERIES OF DATA SAMPLED BY
C   ACQU.FOR AND COMPUTES AND ENSEMBLE
C   AVERAGE.
C
C   THE OUTPUT FILE IS A 4 SEC TIME SERIES
C   (KEEP 100 INFO POINTS, 200Hz data)
C
C   File structure:
C       expAABBB.avg
C           AA: experiment number (01, 02 ...)
C           BBB: number of first file in average
C   Input (example):
C       data limit1/4598,-1,-1,-1,-1/
C       data limit2/4675,-1,-1,-1,-1/
C       data upperlimit /4676/
C       These three lines will process file 4598 to 4675. Other
C       ranges can be specified where the -1 are.  The program
C       stop at file 4676.
C
C   10-11-90: FIRST WRITTEN BY ERIC LAMMARE
C
C   COMPILING AND LINKING NOTES:
C   expavg.obj: expavg.for
C           FL /c /4Ydb /G2 /Fs /Gt   .for
C   expavg.exe: expavg.obj
C       LINK $**, $@, NUL, c:\lib\llibfor7+llibc7 /NOD /NOE;
C
    program expavg
C
    integer*2 i, j, f, flag
    integer*2 data(48100), xdata(0:4900)
    integer*4 header(5), filenum, fexp(3), limit1(5), limit2(5)
    integer*4 count, upperlimit
```

```
      real xreal
      character*80 filename
      data header/130,4900,1,0,2/
      data limit1/4598, -1, -1, -1, -1/
      data limit2/4675, -1, -1, -1, -1/
      data upperlimit/4676/
      xdata(0) = 120
C
      count = 4597
 1000 continue
      do 12 i = 1, 4900
        xdata(i) = 0
   12 continue
      do 10 f = 1, 3
        count = count+1
        flag = 0
        do 5 j = 1, 5
          if (count .ge. limit1(j) .and. count .le. limit2(j)) then
            flag = 99
          endif
          if (count .gt. upperlimit) stop
    5   continue
        if (flag .eq. 0) go to 1000
        filenum = count
        write(filename, 2000) filenum
 2000   format('d:\exp\exp', i5.5,'.dat')
        open(16,file=filename, form='binary', access='direct',
     1        recl=1924, status='old')
        read(16) data
        close(16)
        print *,'Read file number = ',count
C
C   SUM THE THREE FILES
C
        do 15 i = 1, 100
          xdata(i) = xdata(i) + data(i)
   15   continue
C
        do 20 i = 101, 4900
          xdata(i) = xdata(i) + data(16800+i)
   20   continue
   10 continue
C
C   MAKE AVERAGE AND WRITE TO DISK
C
      do 30 i = 1, 4900
        xreal = xdata(i) / 3.0
        xdata(i) = nint(xreal)
   30 continue
      print *
C
      write(filename, 3000) count-2
 3000 format('d:\exp3\e3',i5.5,'.avg')
      open(16,file=filename, form='binary', status='unknown')
      write(16) header
      write(16) xdata
      close(16)
      go to 1000
C
  999 stop
      end
```

program: newbgmat.for

```
$NOTRUNCATE
C
C   PROGRAM NEWBGMAT
C
C   USE TO BUILD BG MATRIX FROM EXP0????.AVG
C   10-13-90: FIRST WRITTEN BY ERIC LAMMARE
C
C   NOTES
C       1) WILL OUTPUT TIME AT EVERY 0.05 sec
C       2) DATA MUST STAY IN WATER DURING THE
C       14SEC TO 18SEC PERIOD IN ORDER TO OBTAIN
C       WATER DC OFFSET. IF DCOFFSET NOT
C       WHITHIN 10% OF WATER CAL. THAN
C       WATER CAL. IS USED.
C
C   Inputs:
C       start.in is an ascii file containing the following
C            start(1) = starting x location of good BG & WG
C            start(2) =  deltax of meas. for BG and WG
C            start(3) = starting y location of good BG
C            start(4) = deltay of meas for BG
C
C   Output: bgAAA.mat where AAA is the wave number
C
C       data tresh/3/ means 3 digital values will be used in the
C       calculation of the DC on the water side
C
C       file1 and file2: starting and ending file # to be processed
C
C       averaging time in samples (always use 10=10samples)
C
C       wt: wave number (either 1, 2 or 3)
C
C   COMPILING AND LINKING NOTES:
C   newbgmat.obj: newbgmat.for
C         FL /c /4Ydb /G2 /Fs /Gt newbgmat.for
C   newbgmat.exe: newbgmat.obj
C       LINK $**, $@, NUL, c:\lib\llibfor7+llibc7 /NOD /NOE;
C
      program newbgmat
C
      integer*4 filenum, i, j, k
      integer*4 file1, file2
      integer*4 sumwater, counter, header(5)
      integer*2 num, max, tresh, time
      integer*2 data(0:4900), shdata(800), tidata(79)
      integer*2 bg(0:45,0:23,1:79), pos(0:45,0:23), out(4:6,6)
      integer*2 limits(4), xloc, yloc
      integer*2 wt
      real vair, vnorm, vf, norm, off, outreal(18)
      real start(4)
      character*80 filename
C
      data tresh/3/
C
C   READ FILE INFORMATION
C
      write(*,'(a\)') ' Enter starting conversion file1: '
      read (*,*) file1
      write(*,'(a\)') ' Enter ending conversion file2: '
      read (*,*) file2
      write(*,'(a\)') ' Averaging time (#sample): '
      read (*,*) time
      write(*,'(a\)') ' Enter wavetype: '
      read (*,*) wt
```

```
C
      write(filename, 1990) wt
 1990 format('d:\eric\start', i3.3,'.in')
      open(16,file=filename,status='old')
      read(16,*) start(1), start(2), start(3), start(4)
      read(16,*) limits(1), limits(2), limits(3), limits(4)
      close(16)
C
C   START LOOP
C
      do 10 k = file1, file2
        if (wt .eq. 1) then
          if (k .gt. 1488 .and. k .lt. 3000) go to 10
        endif
        filenum = k
C
C   ENTER INPUT FILE + READ DATA FILE
C
        if (wt .eq. 1) then
          write(filename, 2000) filenum
 2000     format('d:\exp1\e1', i5.5,'.avg')
        endif
        if (wt .eq. 2) then
          write(filename, 2001) filenum
 2001     format('d:\exp2\e2', i5.5,'.avg')
        endif
        if (wt .eq. 3) then
          write(filename, 2002) filenum
 2002     format('d:\exp3\e3', i5.5,'.avg')
        endif
        open(16, err=10, file=filename, form='binary',
     1          blocksize=512, status='old')
        print *,'Reading filenum = ',filenum
        read(16) header
        read(16) data
        close(16)
        if (wt .eq. 1) then
          write(filename, 2010) filenum
 2010     format('d:\exp1\e1', i5.5,'.out')
        endif
        if (wt .eq. 2) then
          write(filename, 2011) filenum
 2011     format('d:\exp2\e2', i5.5,'.out')
        endif
        if (wt .eq. 3) then
          write(filename, 2012) filenum
 2012     format('d:\exp3\e3', i5.5,'.out')
        endif
        open(16, file=filename)
        read(16,*) (outreal(i), i=1, 18)
        do 12 i = 4, 6
          do 12, j = 1, 6
            out(i,j) = nint(outreal(6*(i-4)+j))
 12     continue
        close(16)
C
C   EXTRACT BG TIME SERIES FROM SAMPLE 2800 TO
3600
C   DO AVERAGING OF 20 SAMPLES TO OBTAIN 50 mS
TIME AVG
C   SECTIONS
C
        do 20 i = 4, 6
          call givepos(wt, start, data(28+2*i), data(29+2*i),
     1                 xloc, yloc)
          if (xloc .lt. limits(1) .or. xloc .gt. limits(2) .or.
     1        yloc .lt. limits(3) .or. yloc .gt. limits(4)) go to 20
          pos(xloc, yloc) = pos(xloc, yloc) + 1
          do 30 j = 0, 799
            shdata(j+1) = data(100+6*j+i)
 30       continue
          call dcoff(shdata, tresh, off)
          if (off .lt. 0.9*0.4096*data(11+2*i)) then
            off = 0.4096*data(11+2*i)
          endif
          vair = 0.4096*data(2*i+10)
          norm = off - vair
          call timeavg(i, out, shdata, time, tidata)
          do 40 j = 1, 79
            if (tidata(j) .eq. 3000) then
              vf = 3
            else
              vnorm = (float(tidata(j)) - vair) / norm
              vf = 0.997 - 1.143*vnorm + 0.1460*vnorm**2
            endif
            bg(xloc,yloc,j) = int( 10000*vf )
 40       continue
 20     continue
 10   continue
      do 50 i = 0, 45
        do 50 j = 0, 23
          do 50 k = 1, 79
            if (pos(i,j) .eq. 0) then
              bg(i,j,k) = 20000
            endif
            if (pos(i,j) .gt. 1) then
              print *,'error, too many position', i, j, pos(i,j)
              stop
            endif
 50   continue
C
C   OUTPUT FILE
C
      write(filename, 2020) wt
 2020 format('d:\eric\bg', i3.3,'.mat')
      open(16,file=filename, form='binary', access='direct',
     1     recl=1024, status='unknown')
      write(16) bg
      close(16)
C
      stop
      end
C
C   SUBROUTINE DCOFF
C
C   USED TO CALCULATE DC OFFSET OF THE WATER
SIDE
C
      subroutine dcoff(shdata, tresh, off)
C
      integer*2 shdata(800)
      integer*2 max, counter, tresh, i
      integer*4 sumwater
      real off
C
      max = -3000
      do 10 i = 1, 800
        if (shdata(i) .gt. max) then
          max = shdata(i)
        endif
 10 continue
   counter = 0
   sumwater = 0
   do 20 i = 1, 800
     if (shdata(i) .gt. max-tresh) then
       counter = counter + 1
       sumwater = sumwater + shdata(i)
     endif
 20 continue
   off = float(sumwater) / float(counter)
C
   end
C
C   SUBROUTINE TIMEAVG
C
C   PERFORMS TIME AVERAGING OF THE SIGNAL
OVER A PERIOD OF time
C   CENTERED AROUND POINTS SPACED EVERY 0.1 S.
C
   subroutine timeavg(g, out, shdata, time, tidata)
C
   integer*4 g
   integer*2 shdata(800), out(4:6,6)
   integer*2 tidata(79), count
   integer*2 i, j, time, space, index
   real sum
C
   space = 5 - time/2
   do 10 i = 1, 79
     sum = 0.0
     count = 0
     do 20 j = 1, time
       index = 5+10*(i-1)+space+j
       if (index .gt. out(g,1) .and. index .lt. out(g,2)) go to 20
       if (index .gt. out(g,3) .and. index .lt. out(g,4)) go to 20
       if (index .gt. out(g,5) .and. index .lt. out(g,6)) go to 20
       count = count + 1
       sum = sum + shdata(index)
 20    continue
     if (count .eq. 0) then
       tidata(i) = 3000
     else
       tidata(i) = int(sum / count)
     endif
 10 continue
C
   end
```

program: wgmat.for

```
$NOTRUNCATE
C
C   PROGRAM WGMAT
C
C   USE TO BUILD WG MATRIX FROM .AVG DATA FILES
C
C   10-13-90: FIRST WRITTEN BY ERIC LAMMARE
C
C   NOTES 1) WILL OUTPUT TIME AT EVERY 0.05 sec
C
C   inputs:
C       file1 and file2: starting and ending file number
C       averaging time 10=10samples or 0.05sec
C       wt: wave number (1, 2 or 3)
C       start: (input file => see newbgmat)
C
C   output: wgAAA.mat where AAA is wave number
C
C   COMPILING AND LINKING NOTES:
C   wgmat.obj: wgmat.for
C           FL /c /4Ydb /G2 /Fs /Gt wgmat.for
C   wgmat.exe: wgmat.obj
C       LINK $**, $@, NUL, c:\lib\llibfor7+llibc7 /NOD /NOE;
C
   program wgmat
C
   integer*4 filenum, i, j, k
   integer*4 file1, file2
   integer*4 header(5)
   integer*2 time
   integer*2 data(0:4900), shdata(800), tidata(79)
   integer*2 wg(0:45,1:79), pos(0:45)
   integer*2 limits(4), xloc, yloc, xpos
   integer*2 wt, off
   real h, a0, a1, a2, a3
   real start(4), height
   character*80 filename
C
C   READ FILE INFORMATION
C
   write(*,'(a\)') ' Enter starting conversion file1: '
   read (*,*) file1
   write(*,'(a\)') ' Enter ending conversion file2: '
   read (*,*) file2
   write(*,'(a\)') ' Averaging time (#sample): '
   read (*,*) time
   write(*,'(a\)') ' Enter wavetype: '
   read (*,*) wt
C
   write(filename, 1990) wt
1990 format('d:\eric\start', i3.3,'.in')
   open(16,file=filename,status='old')
   read(16,*) start(1), start(2), start(3), start(4)
   read(16,*) limits(1), limits(2), limits(3), limits(4)
   close(16)
C
C   START LOOP
C
   do 10 k = file1, file2
     filenum = k
     if (wt .eq. 2) then
       if (k .ge. 2067 .and. k .le. 2240) go to 10
     endif
C
C   ENTER INPUT FILE + READ DATA FILE
C
     if (wt .eq. 1) then
       write(filename, 2000) filenum
2000     format('d:\exp1\e1', i5.5,'.avg')
     endif
     if (wt .eq. 2) then
       write(filename, 2001) filenum
2001     format('d:\exp2\e2', i5.5,'.avg')
     endif
     if (wt .eq. 3) then
       write(filename, 2002) filenum
2002     format('d:\exp3\e3', i5.5,'.avg')
     endif
     open(16, err=10, file=filename, form='binary',
   1        blocksize=512, status='old')
       print *,'Reading filenum = ',filenum
       read(16) header
       read(16) data
       close(16)
C
C   DO AVERAGING OF 20 SAMPLES TO OBTAIN 50 mS
TIME AVG
C   SECTIONS
C
       do 20 i = 1, 3
         xloc = (data(41+i)-start(1))/start(2)
         if (xloc .lt. limits(1) .or. xloc .gt. limits(2)) go to 20
         pos(xloc) = pos(xloc) + 1
         do 30 j = 0, 799
           shdata(j+1) = data(100+6*j+i)
  30     continue
         call avg(shdata, time, tidata)
         a0 = data(20+4*i+0)/1000.0
         a1 = data(20+4*i+1)/1000.0
         a2 = data(20+4*i+2)/1000.0
         a3 = data(20+4*i+3)/1000.0
         off = data(46+i)-a0
         do 40 j = 1, 79
           if (tidata(j) .eq. 3000) then
             height = 300
           else
             h = (tidata(j)-off)/1000.0
             height = a0 + a1*h + a2*h**2 + a3*h**3
           endif
           wg(xloc,j) = wg(xloc,j) + nint(10*height)
  40     continue
  20   continue
  10 continue
     do 50 i = 0, 45
       do 50 k = 1, 79
         if (pos(i) .eq. 0) then
           wg(i,k) = 20000
         else
           wg(i,k) = wg(i,k)/pos(i)
         endif
  50 continue
C
C   OUTPUT FILE
C
     write(filename, 2020) wt
2020 format('d:\eric\wg', i3.3,'.mat')
     open(16,file=filename, status='unknown')
     do 60 i = 0, 45
       do 60 k = 1, 79
         xpos = nint(i*start(2)+start(1))
         write(16,3000) xpos, k, wg(i,k)
3000 format(i6,2x,i6,2x,i6)
  60 continue
     close(16)
C
     stop
     end
C
C   SUBROUTINE TIMEAVG
C
C   PERFORMS TIME AVERAGING OF THE SIGNAL
OVER A PERIOD OF time
C   CENTERED AROUND POINTS SPACED EVERY 0.1 S.
C
```

```
      subroutine avg(shdata, time, tidata)
C
      integer*2 shdata(800)
      integer*2 tidata(79), count
      integer*2 i, j, time, space, index
      real sum
C
      space = 5 - time/2
      do 10 i = 1, 79
        sum = 0.0
        count = 0
        do 20 j = 1, time
          index = 5+10*(i-1)+space+j
          count = count + 1
          sum = sum + shdata(index)
  20    continue
        if (count .eq. 0) then
          tidata(i) = 3000
        else
          tidata(i) = int(sum / count)
        endif
  10  continue
C
      end
```

program:grid.for

```
C
C   PROGRAM GRID
C
C   PROGRAM TO GRID  THE RAW BUBBLE GAUGE
C   DATA
C
C   INPUT : bgmat.dat
C
C   OUTPUT: TIME****.BG CONTAINING GRIDED BG
C   DATA
C
C   inputs:
C       data influ/5/ (# nodes to extend extrapolation)
C       data cay/4/ (smoothing factor)
C       wt: wave number (1, 2 or 3)
C       start: (see newbgmat.for)
C       time1 and time2: starting time and end time to grid
C            use step "timestep"
C       bgAAA.mat file containing the data to grid
C
C    output: *.grd
C
C   Special librarie: drive88.lib and plot88.lib from PLOT88
C   Software, Plotwaorks inc.
C
C   06-22-90: FIRST WRITTEN BY ERIC LAMMARE
C   09-24-90: MODIFIED
C
C   COMPILING ANG LINKING NOTES FOR MICROSOFT
C   FORTRAN 4.1:
C   grid.obj: grid.for
C     FL /c /4Ydb /Zl /G2 /Fs /Gt grid.for
C   grid.exe: grid.obj
C     LINK $**, $@, NUL,c:\lib\llibfor7+llibc7+plot88+drive88
C     /SEG:1024 /NOD /NOE;
C
      program grid
```

```
C
      integer*2 i, j, k, counter, wt, flag
      integer*2 time1, time2, timestep, influ
      integer*2 bg(0:45,0:23,1:79)
      integer*2 limits(4)
      integer*2 wheight, closei
      real xlow, xhigh, ylow, yhigh
      real xp(1104), yp(1104), zp(1104), xpij(1104), knxt(1104)
      real z(0:205,0:55)
      real cay, wg
      real start(4)
      character*30 filename
C
      data influ/5/, cay/4.0/
C
C     READ OPERATOR INPUT
C
      write(*,'(a\)') ' Enter wavetype: '
      read (*,*) wt
      write(filename,1980) wt
 1980 format('start',i3.3,'.in')
      open(16,file=filename,status='old')
      read(16,*) start(1), start(2), start(3), start(4)
      read(16,*) limits(1), limits(2), limits(3), limits(4)
      close(16)
C
      write(*,'(a\)') ' Enter starting time (1,79): '
      read (*,*) time1
      write(*,'(a\)') ' Enter ending time (1,79): '
      read (*,*) time2
      write(*,'(a\)') ' Enter timestep (i.e. time1, time2, timestep): '
      read (*,*) timestep
C     write(*,'(a\)') ' Enter influ (5), cay(4.0): '
C     read (*,*) influ, cay
C
C     READ INPUT FILE
C
      write(filename,1990) wt
 1990 format('bg',i3.3,'.mat')
      open(16,file=filename, form='binary', access='direct',
     1     recl=1024, status='unknown')
      read(16) bg
      close(16)
      print *,'Read bgmat file ok. '
      xlow = start(1)/100.0
      xhigh = start(1)/100.0 + 2.05
      yhigh = start(3)/100.0
      ylow = start(3)/100.0 - 0.55
C
C     EXTRACT TIME TO BE GRIDED
C
      do 10 k = time1, time2, timestep
        do 15 i = 0, 205
          do 15 j = 0, 55
            z(i,j) = 0.0
   15   continue
        counter = 0
        do 20 i = limits(1), limits(2)
          flag = 0
          do 20 j = limits(4), limits(3), -1
            if (bg(i,j,k)/100.0 .gt. 100.0
     1         .and. bg(i,j,k)/100.0 .lt. 250.0) then
              bg(i,j,k) = 0.0
            endif
            if (bg(i,j,k) .lt. 0.0) then
              bg(i,j,k) = 0.0
            endif
            if (bg(i,j,k)/100.0 .gt. 250.0) then
              flag = 1
            endif
            counter = counter + 1
            xp(counter) = float(start(1)+start(2)*i)/100.0
            yp(counter) = float(start(3)-start(4)*j)/100.0
            if (flag .eq. 1) then
              bg(i,j,k) = bg(i,j+1,k)
              zp(counter) = bg(i,j,k)/100.0
            else
              zp(counter) = bg(i,j,k)/100.0
            endif
   20   continue
C       do 22 i = 1, counter
C         print *, xp(i), yp(i), zp(i)
C 22    continue
C
C     USE PLOT88 GRIDING PROGRAM TO GENERATE
DATA
C
        print *,'Grid starts now...'
        call zgrid(z,206,56,206,56,xlow,ylow,xhigh,yhigh,
     1           xp,yp,zp,1104,cay,influ,xpij,knxt)
        print *,'Grid completed.'
        print *,'Smoothing starts now...'
        call zsmth(z,206,56,206,56,4)
        print *,'Smoothing completed.'
C
C     WRITE GRIDED DATA TO D:\GRID\TIME****.WG
C
        write(filename, 2000) wt, k
 2000   format('d:\grid\bg',i3.3,'k',i2.2,'.grd')
        open(16,file=filename,form='binary',access='direct',
     1       recl=1024, status='unknown')
        write(16) z
        close(16)
        print *,'Writing grided file completed:', k
   10 continue
C
      stop
      end
```

program: plotexp.for

```
C
C     PROGRAM PLOTEXP
C
C     PROGRAM TO PLOT BUBBLE CLOUD
C
C     INPUT :   time: specific time to be plotted
C               start.in (see newbgmat.for)
C               fact: scaling factor
C               model: (see below)
C
C     OUTPUT: PLOT TO SCREEN (MODEL 91) OR
C     PAINTJET XL (MODEL 68)
C
C     06-01-90: FIRST WRITTEN BY ERIC LAMMARE
C     06-20-90: MODIFIED
C     06-22-90: MODIFIED
C     07-31-90: MODIFIED
```

```
C   10-25-90: MOFIFIED FOR FALL 90 BG EXPERIMENT
C
C   Special libraries: drive88.lib and plot88.lib from PLOT88
C   software, Plotworks inc.
C
C   COMPILING AND LINKING NOTES FOR
C   MICROSOFT FORTRAN 4.1:
C   plotexp.obj: plotexp.for
C       FL /c /4Ydb /Zl /G2 /Fs plotexp.for
C   box.obj: box.for
C       FL /c /4Ydb /Zl /G2 /Fs box.for
C   asscol.obj: asscol.for
C       FL /c /4Ydb /Zl /G2 /Fs asscol.for
C   legend.obj: legend.for
C       FL /c /4Ydb /Zl /G2 /Fs legend.for
C   axisp.obj: axisp.for
C       FL /c /4Ydb /Zl /G2 /Fs axisp.for
C   plotexp.exe: plotexp.obj box.obj asscol.obj legend.obj
C   axisp.obj
C   LINK $**, $@, NUL, c:\lib\plot88+drive88+llibfor7+llibc7
C   /SEG:1024 /NOD /NOE;
C
      program plotexp
C
      integer*4 i, j, k, err, nc, countx, county, ii, dum
      integer*4 ioport, model
      integer*4 time, col, col2(20), colarray(20)
      integer*2 start(4), limits(4), wt
      integer*2 wg1(0:45,1:79), wg2(0:45,1:79)
      real z(0:205,0:55)
      real rtime
      real fact, voidf
      real xpos, ypos, wxpos, wypos, pos, f, waveh, dist
      real sumarea, sumenergy, sumair, meanvoid
      real window(4), xtick, ytick, height
      real range(20)
      real dx, dy, area
      real xlow, ylow, xhigh, yhigh
      character*10 ctext
      character*30 filename
C
      data xtick/0.1/, ytick/0.1/, height/0.060/
      data ioport/1/
      data colarray /8,4,13,6,2,9,12,1,7,3,
     1           15,15,15,15,15,15,15,15,15,15/
      data col2 /5,1,9,11,2,10,4,13,12,14,
     1        15,15,15,15,15,15,15,15,15,15/
      data range/0.3,0.5,1.0,3.0,5.0,10.0,15.0,20.0,40.0,60.0,100.0,
     1
100.0,100.0,100.0,100.0,100.0,100.0,100.0,100.0,100.0/
C
C   INPUT FILE # TO BE PLOTED, MODEL AND SIZE
FACTOR AND WINDOW
C
      write(*,'(a\)') ' Enter wavetype: '
      read (*,*) wt
C
      write(filename,1980) wt
 1980 format('start',i3.3,'.in')
      open(16,file=filename,status='old')
      read(16,*) start(1), start(2), start(3), start(4)
      read(16,*) limits(1), limits(2), limits(3), limits(4)
      close(16)
C
      print *,' Enter time, model, fact:'
      read(*,*) time, model, fact
C
      rtime = float(time) * 0.05 + 14.0
      if (model .eq. 91) then
        ioport = 91
        do 3 i = 1, 20
          colarray(i) = col2(i)
  3     continue
      endif
      write(*,'(a\)') ' Enter window
(wxlow,wylow,wxhigh,wyhigh): '
      read (*,*) window(1), window(2), window(3), window(4)
      write(*,'(a\)') ' Enter sumarea, sumair, sumenergy: '
      read (*,*) sumarea, sumair, sumenergy
      xlow = start(1)/100.0
      xhigh = start(1)/100.0 + 2.05
      yhigh = start(3)/100.0
      ylow = start(3)/100.0-0.55
      write(filename, 2022) wt
 2022 format('d:\eric\wg', i3.3,'.mat')
      open(16,file=filename, status='old')
      do 62 i = 0, 45
        do 62 k = 1, 79
          read(16,3002) dum, dum, wg1(i,k)
 3002 format(i6,2x,i6,2x,i6)
  62  continue
      close(16)
C
      write(filename, 2021) wt
 2021 format('d:\eric\wg', i3.3,'.mod')
      open(16,file=filename, status='old')
      do 61 i = 0, 45
        do 61 k = 1, 79
          read(16,3001) dum, dum, wg2(i,k)
 3001 format(i6,2x,i6,2x,i6)
  61  continue
      close(16)
C
C
C   READ INPUT FILES (GENERATED BY GRID.FOR)
C
      write(filename, 2000) wt, time
 2000 format('d:\grid\bg',i3.3,'k',i2.2,'.grd')
      open(16,file=filename,form='binary',access='direct',
     1     recl=1024, status='old')
      read(16) z
      close(16)
      print *,'Read grided file ok.'
C
C   COMPUTE 1) FRACTIONAL AREA OF AIR
C           2) AREA OF BUBBLE CLOUD
C           3) POTENTIAL ENERGY
C
C   area=1.0
C   call stat(z, wg2, area, time, start, limits,
C   1        sumair, sumarea, sumenergy)
      meanvoid = 100 * (sumair / sumarea)
      print *,'Computation of cloud statistics ok.'
C
C   INITIALIZE PLOT88 LIBRARY, SET FACTOR SIZE,
DRAW BOX AND LABELS
C
      call plots(0, ioport, model)
      call palete(12,0.302,0.583,0.127)
      call palete(9,0.070,0.405,0.203)
      call plot(1.0,1.6,-3)
      call factor( fact )
      xpos = (window(3)-window(1))/2.0 - 13*height/fact
      ypos = (window(4)-window(2)) + 2.0*height/fact
*     call symbol(xpos,ypos,2*height/fact,'air fraction
(%)',0.0,16)
      call symbol(2*height/fact,window(4)-window(2)-
4*height/fact,
     1          2*height/fact,'d)',0.0,2)
C
C     PLOT AXIS
C
      call axisp(window,fact,xtick,ytick,height)
      print *,'Plot88 Initialization ok.'
C
C     PRINT COLOR SPECTRUM LEGEND
C
*     call legend (window,colarray,range,fact,height)
*     print *,'Plot legend ok.'
C
C     PLOT WAVE GAUGE HERE IF MODEL 68
C
      if (model .eq. 68) then
        call color(0,err)
        call wgp(time,0.05,wg1,window,start,limits)
        call wgp(time,-0.05,wg2,window,start,limits)
        print *,'Wave gauge plotted.'
C
C     PLOT TIME, MEAN VOID FRACTION, AREA OF
CLOUD AND ENERGY
C
        call color(0,err)
C       xpos = (window(3)-window(1))/20.0
        xpos = (window(3)-window(1))*0.75
        ypos = (window(4)-window(2))/6.0
        call symbol(xpos,ypos,1.5*height/fact,'t / T  =',0.0,9)
        call getnum((rtime-14.4)/1.136,2,ctext,nc)
        call symbol(xpos+13*height/fact,ypos,
     1           1.5*height/fact,ctext,0.0,nc)
C
        call symbol(xpos,ypos-2.0*height/fact,1.5*height/fact,
     1           't (sec) =',0.0,9)
        call getnum((rtime-14.4),2,ctext,nc)
        call symbol(xpos+13*height/fact,ypos-2.0*height/fact,
     1           1.5*height/fact,ctext,0.0,nc)
C
C       xpos = (window(3)-window(1))*0.75
C       call symbol(xpos,ypos+2.0*height/fact,1.5*height/fact,
C    1           'mean v.f. (%) =',0.0,15)
C       call getnum(meanvoid,1,ctext,nc)
C       call symbol(xpos+20*height/fact,ypos+2.0*height/fact,
C    1           1.5*height/fact,ctext,0.0,nc)
C
C       call symbol(xpos,ypos,1.5*height/fact,'area (cm^2)
=',0.0,13)
C       call getnum(sumarea,0,ctext,nc)
C       call symbol(xpos+19*height/fact,ypos,
C    1           1.5*height/fact,ctext,0.0,nc)
C
C       call symbol(xpos,ypos-2.0*height/fact,1.5*height/fact,
C    1           'energy (J/m) =',0.0,14)
C       call getnum(sumenergy,1,ctext,nc)
C       call symbol(xpos+20*height/fact,ypos-2.0*height/fact,
C    1           1.5*height/fact,ctext,0.0,nc)
C       print *,'Plot statistics ok.'
      endif
C
C     FIND dx AND dy OF THE BOXES
C
      countx = 0
      county = 0
      do 10 i = 0, 205
        xpos = i*0.01 + xlow
        if (xpos .ge. window(1) .and.
     1      xpos .le. window(3)) then
          countx = countx + 1
        endif
 10   continue
      do 15 j = 0, 55
        ypos = j*0.01 + ylow
        if (ypos .ge. window(2) .and.
     1      ypos .le. window(4)) then
          county = county + 1
        endif
 15   continue
      dx = (window(3)-window(1)) / countx
      dy = (window(4)-window(2)) / county
C
C     PLOT THE BOXES
C
      do 20 i = 0, 205
        if (i .eq. 0) print *,'0  % completed'
        if (i .eq. 45) print *,'20 % completed'
        if (i .eq. 90) print *,'40 % completed'
        if (i .eq. 135) print *,'60 % completed'
        if (i .eq. 180) print *,'80 % completed'
        do 20 j = 0, 55
          xpos = i*(xhigh-xlow)/205 + xlow
          ypos = j*(yhigh-ylow)/55 + ylow
          if (xpos .le. window(1) .or. xpos .ge. window(3) .or.
     1        ypos .le. window(2) .or. ypos .ge. window(4)) then
            go to 20
          endif
          wxpos = xpos - window(1)
          wypos = ypos - window(2)
C
          pos = float(i)/float(start(2))
          ii = int(pos)
          f = pos-ii
          waveh = ((1-f)*wg2(ii,time) + f*wg2(ii+1,time))/10.0
          dist = waveh-(start(3)-55+j)
          if (dist .lt. 3.0) then
            voidf = 0.0
          else
            voidf = z(i,j)
          endif
          call asscol(colarray,range,voidf,col)
          call box(wxpos,wypos,dx,dy,col)
 20   continue
      print *,'Plot boxes ok.'
C
C     PLOT WAVE GAUGE HERE IF MODEL 91
C
      if (model .eq. 91) then
        call color(0,err)
        call wgp(time,0.05,wg1,window,start,limits)
        call wgp(time,-0.05,wg2,window,start,limits)
        print *,'Wave gauge plotted.'
C
```

```
C   PLOT TIME, MEAN VOID FRACTION, AREA OF
CLOUD AND ENERGY
C
      call color(0,err)
C       xpos = (window(3)-window(1))/20.0
      xpos = (window(3)-window(1))*0.75
      ypos = (window(4)-window(2))/10.0
      call symbol(xpos,ypos,1.5*height/fact,'t / T  =',0.0,9)
      call getnum((rtime-14.4)/1.136,2,ctext,nc)
      call symbol(xpos+11*height/fact,ypos,
   1          1.5*height/fact,ctext,0.0,nc)
C
      call symbol(xpos,ypos-2.0*height/fact,1.5*height/fact,
   1          't (sec) =',0.0,9)
      call getnum((rtime-14.4),2,ctext,nc)
      call symbol(xpos+11*height/fact,ypos-2.0*height/fact,
   1          1.5*height/fact,ctext,0.0,nc)
C
C     call symbol(xpos,ypos,1.5*height/fact,'area (cm^2)
=',0.0,13)
C     call getnum(sumarea,0,ctext,nc)
C     call symbol(xpos+19*height/fact,ypos,
C  1          1.5*height/fact,ctext,0.0,nc)
C
C     call symbol(xpos,ypos-2.0*height/fact,1.5*height/fact,
C  1          'energy (J/m) =',0.0,14)
C     call getnum(sumenergy,1,ctext,nc)
C     call symbol(xpos+20*height/fact,ypos-2.0*height/fact,
C  1          1.5*height/fact,ctext,0.0,nc)
C     print *,'Plot statistics ok.'
    endif
C
C   END OF PLOT
C
    call plot(0.0,0.0,999)
C
    stop
    end
C
C   SUBROUTINE WGP
C
C   PLOTS WAVE GAUGE
C
    subroutine wgp(time,linetype,wg,window,start,limits)
C
    integer*2 wg(0:45,1:79), limits(4), i, count, finalcount
    integer*2 start(4)
    integer*4 time
    real window(4), xarray(0:45), yarray(0:45), xpos
    real linetype
C
    count = -1
    do 10 i = limits(1), limits(2)
      xpos = (start(1) + i*start(2))/100.0
      if (xpos .lt. window(1)+0.05 .or. xpos .gt. window(3)) go to
10
      count = count + 1
      xarray(count) = xpos-window(1)
      if (wg(i,time) .eq. 20000) then
        wg(i,time) = (wg(i-1,time) + wg(i+1,time))/2.0
      endif
      yarray(count) = wg(i,time)/1000.0-window(2)
  10 continue
  finalcount = count
    do 20 i = finalcount+1, 45
      xarray(i) = xarray(finalcount)
      yarray(i) = yarray(finalcount)
  20 continue
    do 21 i = 0, 45
  21 continue
    call curve(xarray, yarray, -46, linetype)
C
    end
C
C   SUBROUTINE STAT
C
C   SUBROUTINE TO COMPUTE STATISTICS ABOUT
BUBBLE CLOUD
C
C   INPUT : Z ARRAY
C           WG
C
C   OUTPUT: RETURNS VALUE OF 1) SUMAIR - FRACT.
AREA OF AIR
C                   2) SUMAREA - BUBBLE CLOUD AREA
C                   3) SUMENERGY - POTENTIAL ENERGY
C
C   06-22-90: FIRST WRITTEN BY ERIC LAMMARE
C   10-25-90: MODIFIED
C
    subroutine stat(z, wg, area, k, start, limits,
   1          sumair, sumarea, sumenergy)
C
    integer*4 k, i, j, ii
    integer*2 start(4), limits(4)
    integer*2 wg(0:45,1:79)
    real z(0:205,0:55)
    real waveh, f
    real fact, voidf, fracvoidf, pos
    real sumarea, sumenergy, sumair, area, dist
C
    sumarea = 0.0
    sumair = 0.0
    sumenergy = 0.0
C
    do 20 i = 0, 205
      do 20 j = 0, 55
        if (start(1)+i .gt. start(1)+limits(2)*start(2)) go to 20
        voidf = z(i,j)
        pos = i/start(2)
        ii = int(pos)
        f = pos-ii
        waveh = ((1-f)*wg(ii,k) + f*wg(ii+1,k))/10.0
        dist = waveh-(start(3)-55+j)
        if (dist .lt. 0.0) then
          voidf = 0.0
        endif
        fracvoidf = voidf / 100.0
        if (voidf .gt. 0.3 .and. voidf .lt. 100.0) then
          sumair = sumair + fracvoidf * area
          sumarea = sumarea + area
          sumenergy = sumenergy + fracvoidf * (dist/100) *
   1          (area/10000.0) * 9.81 * 1000.0
        endif
  20 continue
C
    end
```

subroutine: box.for

```
C
C   SUBROUTINE BOX
C
C   SUBROUTNIE TO FILL A SQUARE BOX
C
C   INPUT : XPOS, YPOS - LOCATION OF CENTER
C   POINT OF BOX
C   XSIDE, YSIDE - DIMENSIONS OF THE BOX
C   COL - COLOR OF THE BOX
C
C   OUTPUT: PLOTS BOX
C
      subroutine box(xpos,ypos,xside,yside,col)
C
      integer col, err
      real xbox(4), ybox(4)
      real xpos, ypos, xside, yside
C
C   COMPUTES THE VERTICES FOR THE BOX TO BE
DRAWNED
C
      xbox(1) = xpos - xside/2
      ybox(1) = ypos - yside/2
      xbox(2) = xpos + xside/2
      ybox(2) = ypos - yside/2
      xbox(3) = xpos + xside/2
      ybox(3) = ypos + yside/2
      xbox(4) = xpos - xside/2
      ybox(4) = ypos + yside/2
C
C   PLOTING
C
      call color(col,err)
      call fill(xbox,ybox,4)
C
      end
```

subroutine: asscol.for

```
C
C   SUBROUTINE ASSCOL
C
C   SUBROUTINE TO ASSIGN APPROPRIATE COLOR
C   TO VOID FRACTION VALUES
C
C   INPUT : COLARRAY - COLOR ORDER INDICES
C   DEFINE IN MAIN
C   VOID - VOID FRACTION (%) COMPUTED IN MAIN
C
C   OUTPUT: COL - COLOR INDEX RETURNED
C
      subroutine asscol(colarray,range,void,col)
C
      integer col, colarray(20)
      real range(20)
      real void
C
      col = 15
      if (void .gt. range(1) .and. void .le. range(2)) then
        col = colarray(1)
      endif
      if (void .gt. range(2) .and. void .le. range(3)) then
        col = colarray(2)
      endif
      if (void .gt. range(3) .and. void .le. range(4)) then
        col = colarray(3)
      endif
      if (void .gt. range(4) .and. void .le. range(5)) then
        col = colarray(4)
      endif
      if (void .gt. range(5) .and. void .le. range(6)) then
        col = colarray(5)
      endif
      if (void .gt. range(6) .and. void .le. range(7)) then
        col = colarray(6)
      endif
      if (void .gt. range(7) .and. void .le. range(8)) then
        col = colarray(7)
      endif
      if (void .gt. range(8) .and. void .le. range(9)) then
        col = colarray(8)
      endif
      if (void .gt. range(9) .and. void .le. range(10)) then
        col = colarray(9)
      endif
      if (void .gt. range(10) .and. void .le. range(11)) then
        col = colarray(10)
      endif
      if (void .gt. range(11) .and. void .le. range(12)) then
        col = colarray(11)
      endif
      if (void .gt. range(12) .and. void .le. range(13)) then
        col = colarray(12)
      endif
      if (void .gt. range(13)) then
        col = colarray(13)
      endif
C
      end
```

subroutine: legend.for

```
C
C   SUBROUTINE LEGEND
C
C   SUBROUTINE TO PRINT COLOR SPECTRUM LEGEND
C
C   INPUT : COLARRAY - COLOR INDICES ORDER
C   DEFINE IN MAIN
C
C   OUTPUT: PLOTS LEGEND
C
      subroutine legend(window,colarray,range,fact,height)
C
      integer err, colarray(20)
      integer i, count, nc
      real window(4), height, dx, dy, range(20), ypos, fact
      character*5 ctext
C
C   INDENTIFY HOW MANY SQUARES TO BE DRAWN
C
      count = 0
      do 10 i = 20, 1, -1
        if (range(i) .eq. 100.0) then
          count = count+1
        else
```

```
          go to 20
        endif
   10 continue
   20 continue
      dx = (window(3)-window(1))/(20-count)
      ypos = (window(4)-window(2)) + 7*height/fact
      dy = 3*height/fact
      do 30 i = 1, 20-count
        call box((i-1)*dx+dx/2.0,ypos,dx,dy,colarray(i))
        call getnum(range(i),1,ctext,nc)
        call color(0,err)
        call symbol((i-1)*dx+dx/2.0-3*height/fact,
     1        ypos+3.0*height/fact, 1.5*height/fact,ctext,0.0,5)
   30 continue
C
      end
```

subroutine: axisp.for

```
C
C
C     SUBROUTINE AXISP
C
C     OUTPUT:  PLOTS AXIS
C              PLOTS LABELS EVERY 10cm
C              PLOTS AXIS LABELS
C
      subroutine axisp(window,fact,xtick,ytick,height)
C
      integer err, nc, i, intdec
      real decimeter, xtick, ytick, height
      real window(4), xpos, ypos, fact
      character*6 ctext
C
C     PLOT AXIS
C
      call color(0,err)
      call plot(0.0, 0.0, 3)
      call plot(0.0, window(4)-window(2), 2)
      call plot(window(3)-window(1), window(4)-window(2), 2)
      call plot(window(3)-window(1), 0.0, 2)
      call plot(0.0, 0.0, 2)
C
C     PLOT AXIS TICK MARKS EVERY xtick(m) AND
ytick(m)
C     TICK MARKS ARE height HIGH, TICK LABELS ARE
1.5*height HIGH
C     X AXIS
      decimeter = (window(3)-window(1)) / (xtick * 1.936)
      intdec = int(decimeter)
      do 10 i = 0, intdec
        xpos = i * xtick * 1.936
        ypos = 0.0
        call symbol(xpos, ypos, 0.75*height/fact, char(3),0.0,-1)
*        call getnum((window(1)+xpos-8.05)/1.936,2,ctext,nc)
*        call symbol(xpos-3*height/fact, ypos-3*height/fact,
*    1           1.5*height/fact, ctext,0.0,nc)
        ypos = window(4)-window(2)
        call symbol(xpos, ypos, 0.75*height/fact, char(3),0.0,-1)
   10 continue
C     Y AXIS
      decimeter = (window(4)-window(2)) / (ytick*1.936)
      intdec = int(decimeter)
      do 20 i = 0, intdec
        xpos = 0.0
        ypos = i * ytick * 1.936
        call symbol(xpos,ypos,0.75*height/fact,char(3),90.0,-1)
*        call getnum((window(2)+ypos)/1.936,1,ctext,nc)
*        call symbol(xpos-2*height/fact, ypos-2*height/fact,
*    1           1.5*height/fact, ctext,90.0,nc)
        xpos = window(3)-window(1)
        call symbol(xpos,ypos,0.75*height/fact,char(3),90.0,-1)
   20 continue
C
C     PLOT AXIS LABEL WITH 2*height
C
      xpos = (window(3)-window(1))/2.0
      ypos = 0.0
*     call symbol(xpos-3*height/fact, ypos-7*height/fact,
*    1        2*height/fact, 'x / l',0.0,5)
      xpos = 0.0
      ypos = (window(4)-window(2))/2.0
*     call symbol(xpos-5*height/fact,ypos-4*height/fact,
*    1        2*height/fact, 'z / l',90.0,5)
      end
```

program: sumstatb.for

```
C
C     PROGRAM SUMSTATB
C
C     PROGRAM TO COMPUTE BUBBLE CLOUD STAT
C     FOR ALL TIMES
C
C     INPUT:     bg???.MAT file
C                wg???.MAT file
C                wt: wave number (1,2 0r 3)
C                k1, k2: times to be analysed
C                start.in: see newbgmat.for
C                area: dx by dy of measurements
C
C     OUTPUT: SUMSTATB.DAT CONTAINING BUBBLE
C     CLOUD STAT
C
C     06-22-90: FIRST WRITTEN BY ERIC LAMMARE
C
C     COMPILING AND LINKING NOTES FOR MICROSOFT
C     FORTRAN 4.1:
C     sumstat.obj: sumstat.for
C           FL /c /4Ydb /Zl /G2 /Fs sumstat.for
C     statb.obj: stat.for
C           FL /c /4Ydb /Zl /G2 /Fs stat.for
C     sumstat.exe: sumstat.obj statb.obj
C        LINK $**, $@, NUL, c:\lib\llibfor7+llibc7 /SEG:1024
C        /NOD /NOE;
C
      program sumstatb
C
      integer*4 time, dum
      integer*2 i, wt, limits(4), k, k1, k2
      integer*2 bg(0:45,0:23,1:79), wg(0:45,1:79)
      real start(4), sumair, sumarea, sumenergy, meanvoid, area
      real centx, centy, sumcentx
      character*20 filename
C
C     INPUT
C
      print *,' Enter wt:'
      read(*,*) wt
      print *,' Enter k1:'
      read(*,*) k1
      print *,' Enter k2:'
      read(*,*) k2
C
C   READ INPUT FILES
C
      write(filename, 1990) wt
 1990 format('d:\eric\start', i3.3,'.in')
      open(16,file=filename,status='old')
      read(16,*) start(1), start(2), start(3), start(4)
      read(16,*) limits(1), limits(2), limits(3), limits(4)
      close(16)
C
      if (wt .eq. 1) then
        area = 25
      endif
      if (wt .eq. 2) then
        area = 10
      endif
      if (wt .eq. 3) then
        area = 15
      endif
C
      write(filename, 2021) wt
 2021 format('d:\eric\wg', i3.3,'.mod')
      open(16,file=filename, status='old')
      do 61 i = 0, 45
        do 61 k = 1, 79
          read(16,3001) dum, dum, wg(i,k)
 3001 format(i6,2x,i6,2x,i6)
   61 continue
      close(16)
C
      write(filename, 2000) wt
 2000 format('d:\eric\bg', i3.3,'.mat')
      open(16,file=filename ,form='binary', status='old')
      read(16) bg
      close(16)
      print *,'Read file ok.'
C
C   COMPUTE 1) FRACTIONAL AREA OF AIR
C           2) AREA OF BUBBLE CLOUD
C           3) POTENTIAL ENERGY
C
      write(filename, 2020) wt
 2020 format('d:\eric\sumb', i3.3,'.dat')
      open(26,file=filename,status='unknown')
      do 10 time = k1, k2
        print *, time
        call statb(time,bg,wg,start,limits,area,sumair,sumarea,
     1            sumenergy,sumcentx)
        if (sumarea .eq. 0.0) then
          meanvoid = 0.0
        else
          meanvoid = 100 * (sumair / sumarea)
        endif
        centy = 100 * (sumenergy / (sumair/10000) / (9.81*1000))
        centx = sumcentx / sumair
        write(26,1000) 14.0+time*0.05, sumair, sumarea,
     1            sumenergy, meanvoid, centx, centy
 1000   format(7(f9.2,2x))
   10 continue
      close(26)
C
      stop
      end
```

subroutine: statb.for

```
C
C   SUBROUTINE STATB
C
C   SUBROUTINE TO COMPUTE STATISTICS ABOUT
C   BUBBLE CLOUD
C
C   INPUT :   BGMAT ARRAY
C             WGMAT ARRAY
C
C   OUTPUT: RETURNS VALUE OF
C      1) SUMAIR - FRACT. AREA OF AIR
C      2) SUMAREA - BUBBLE CLOUD AREA
C      3) SUMENERGY - POTENTIAL ENERGY
C
C   10-14-90: FIRST WRITTEN BY ERIC LAMMARE
C
      subroutine statb(time,bg,wg,start,limits,area,sumair,sumarea,
     1            sumenergy,sumcentx)
C
      integer*4 i, j, time
      integer*2 bg(0:45,0:23,1:79), wg(0:45,1:79), limits(4)
      real fact, voidf, fracvoidf, wgh, ypos, start(4)
      real sumarea, sumenergy, sumair, area, dist, yhigh
      real xpos, sumcentx, xlow
C
      sumarea = 0.0
      sumair = 0.0
      sumenergy = 0.0
      sumcentx = 0.0
C
      do 20 i = limits(1), limits(2)
        do 20 j = limits(3), limits(4)
          xlow = start(1)/100.0
          xpos = i*start(2)/100.0 + xlow
          yhigh = start(3)/100.0
          ypos = -j*start(4)/100.0 + yhigh
          if(wg(i,time) .eq. 20000) then
            wg(i,time) = (wg(i-1,time)+wg(i+1,time))/2.0
          endif
          wgh = wg(i,time)/1000.0
          dist = wgh - ypos
          voidf = bg(i,j,time)/100.0
          if (dist .lt. 0.0) then
            voidf = 0.0
          endif
          fracvoidf = voidf / 100.0
          if (voidf .gt. 0.3 .and. voidf .le. 100.0) then
            sumair = sumair + fracvoidf * area
            sumarea = sumarea + area
            sumenergy = sumenergy + fracvoidf * dist *
     1          (area/10000.0) * 9.81 * 1000.0
            sumcentx = sumcentx + fracvoidf * (area) * xpos
          endif
   20 continue
C
      end
```

program: swade.for

```
      interface to subroutine com1op [C] ()
      end
      interface to subroutine tcode [C] (time)
      character*16 time [REFERENCE]
      end
C
C     PROGRAM SWADE
C
C     PROGRAM TO ACQUIRE 3 CHANNELS OF DATA
C     AT 200Hz FOR 5 MINUTES
C     PROGRAM USED TO ACQUIRE DATA DURING
C     THE SWADE EXPERIMENT.  DATA IS READY
c     TO BE LOADED IN MATLAB.
C
C     special libraries:
C         gfcl.lib: Greeleaf inc communication library
C         das16f.lib: Metrabyte corp. for their das16 board
C
C     special functions:
C         com1open.obj: C function to open serial port
C         tcode.obj: C function to read Datum time-code
C
C     01-06-91: FIRST WRITTEN BY ERIC LAMARRE
C
C     COMPILING AND LINKING NOTES
C     swade.obj: swade.for
C       FL /c /4Ydb /G2 /4I2 /Fs swade.for
C     swade.exe: swade.obj com1open.obj tcode.obj
C       LINK $**, $@, NUL, c:\lib\llibfor7+llibc7+gfcl+das16f
C       /NOD /NOE;
C
      program swade
C
      integer*2 baddr
      integer*2 base
      integer*2 buf, halfbuf
      integer*2 buffer[far]
      integer*2 ch
      integer*2 cycpulse
      integer*2 das16
      integer*2 data(4050)
      integer*2 dmalev
      integer*2 fch
      integer*4 filenum
      integer*4 header(5)
      integer*2 i, j, k, l, m, p, ierr
      integer*2 ihr(2), imin(2), isec(2), i100th(2)
      integer*2 idecode
      integer*2 intlev
      integer*4 longdelay, shortdelay
      integer*2 mode0, mode1, mode6, mode7, mode8, mode9,
mode17
      integer*2 nbuf
      integer*2 nch
      integer*2 ncyc
      integer*2 n1rate, n2rate
      integer*2 oaddr
      integer*2 offadr
      integer*2 param(5)
      integer*2 rcode
      integer*2 rcyc
      integer*2 samples
      integer*2 segadr
      integer*2 sch
      integer*2 trig
      integer*2 varname
C
      real dig
C
      character*20 filename
      character*1 flag
      character*16 time1, time2
C
      dimension buffer(32767)
C
      data base/#340/
      data buf/8100/
      data dmalev/1/
      data header/130,59400,3,0,2/
      data intlev/3/
      data longdelay/714285/, shortdelay/3600/
C      714285 = 2 Sec, 3600 = 10mSec
      data mode1/1/, mode0/0/, mode6/6/, mode7/7/, mode8/8/
      data mode9/9/, mode17/17/
C      ncyc = 22 for 5 min. record (with 8100 buf)
      data ncyc/22/
      data n1rate/166/, n2rate/10/
      data rcyc/1/
      data sch/0/, fch/2/
      data time1/'  : : : .  '/
      data time2/'  : : : .  '/
      data trig/1/
      varname = 120
C
C      TEST FOR SEATTLE
C
      write(*,*) time1
      call com1open
      call tcode(time1)
      write(*,*) time1
      stop
C
C
C      INPUT OF PARAMETERS + COMPUTATION OF
OTHER PARAMETERS
C
      nch = fch - sch + 1
      halfbuf = buf / 2
C
C      OPEN OUTPUT FILE AND WRITE MATLAB HEADER
C
      open(16,file='swade.in',status='old')
      read(16,*) filenum
      close(16)
      write(filename,40) filenum+1
   40 format('d:\swd\swd',i5.5,'.dat')
      open(16,file=filename,form='binary',status='new')
      write(*,45) filename
   45 format(' File: ',a20,' opened.')
      write(16) header
      write(16) varname
C
C      INITIALIZE SERIAL PORT COM1 FOR INTERFACE
WITH TIME CODE
C
      call com1open
C
C      INITIALIZE DAS-16 A/D BOARD
```

```
C
     param(1) = base
     param(2) = intlev
     param(3) = dmalev
     rcode = das16(mode0, param)
     if(rcode .ne. 0) print *,' Error mode0, rcode = ',rcode
C
C    SET CHANNEL SCAN LIMITS
C
     param(1) = sch
     param(2) = fch
     rcode = das16(mode1, param)
     if(rcode .ne. 0) print *,' Error mode1, rcode = ',rcode
C
C    SET CLOCK RATE
C
     param(1) = n1rate
     param(2) = n2rate
     rcode = das16(mode17, param)
     if(rcode .ne. 0) print *,' Error mode17, rcode = ',rcode
C
C    SET UP BUFFER POINTER
C
     baddr = segadr(buffer)
C
C    BEGIN DMA CONVERSION TO buffer
C
     param(1) = buf
     param(2) = baddr
     param(3) = trig
     param(4) = rcyc
C    pause 'Press return to begin sampling'
     write(*,46)
  46 format(' Sampling begins ...')
     print *
     call pulse(longdelay)
     call tcode(time1)
     rcode = das16(mode6, param)
     call pulse(shortdelay)
     call gettim(ihr(1), imin(1), isec(1), i100th(1))
     if(rcode .ne. 0) print *,' Error mode6, rcode = ',rcode
C
C    CHECK IF HALF BUFFER FILLED, IF YES INITIATE
TRANSFER
C    FROM buffer TO data ARRAYS
C
     cycpulse = 0
     do 10 i = 1, ncyc
       write(*,1000) i
 1000  format('+',i2,'/22')
       do 10 j = 1, 2
  100    rcode = das16(mode8, param)
         do 30 l = 1, 100
   30    continue
         if (j .eq. 1 .and. param(3) .le. halfbuf .or.
    1      j .eq. 2 .and. param(3) .gt. halfbuf) then
           go to 100
         else
           if (i .eq. ncyc .and. j .eq. 2) then
             call gettim(ihr(2), imin(2), isec(2), i100th(2))
             call tcode(time2)
           endif
           do 20 k = 1, halfbuf
             data(k) = idecode(buffer(halfbuf*(j-1)+k))
   20      continue
           write(16) data
         endif
       cycpulse = cycpulse + 1
       if (cycpulse .eq. 2) then
         call pulse(shortdelay)
         cycpulse = 0
       endif
  10 continue
C
C    SHUT DOWN DMA
C
     rcode = das16(mode7, param)
     call pulse(longdelay)
     if(rcode .ne. 0) print *,' Error mode7, rcode = ',rcode
     close(16)
     write(*,47)
  47 format(' Sampling terminated.')
     write(*,2000) ihr(1), imin(1), isec(1), i100th(1)
     write(*,2000) ihr(2), imin(2), isec(2), i100th(2)
2000 format(i2.2,':',i2.2,':',i2.2,':',i2.2)
C
C    WRITING LOG INFO TO swd?????.log
C
     write(filename,50) filenum+1
  50 format('d:\log\swd',i5.5,'.log')
     open(16,file=filename,status='new')
     write(16,*) filenum+1
     write(16,*) ihr(1), imin(1), isec(1), i100th(1)
     write(16,*) ihr(2), imin(2), isec(2), i100th(2)
     write(16,*) time1
     write(16,*) time2
     close(16)
C
C    WRITE NEW FILENUM TO DISK
C
     open(16,file='swade.in',status='old')
     write(16,*) filenum+1
     close(16)
C
     stop
     end
C
C    FUNCTION IDECODE
C
C    CONVERTS 12 BITS DATA INTO 16 BITS DATA
C
     integer*2 function idecode(ibin)
       integer*2 ibin
       idecode = isha(ibin, -4) - sign(2048, ibin)
     end
C
C    SUBROUTINE PULSE
C
C    SEND SHORT PULSE TO DIG. OUT 0
C    THIS PULSE GOES TO ANALOG SWITCH IN ACQU.
BOX TO GENERATE
C    A SMALL DEAD SIGNAL ON HYDROPNONE
CHANNEL
C
     subroutine pulse(delay)
C
     integer*4 i
     integer*4 delay
     integer*2 param(5)
     integer*2 mode13
```

```
      integer*2 rcode, das16
C
      data mode13/13/
C
      param(1) = 1
      rcode = das16(mode13, param)
      do 10 i = 1, delay
   10 continue
      param(1) = 0
      rcode = das16(mode13, param)
C
      end
```

function: com1open.c

```
/***************************************************
funtion com1open.c

Function to open the com port for serial communication.
This function is to be used in Fortran programs to communicate
with the Datum time code generator.

from Jessup [1990]

***************************************************/

#include <stdio.h>
#include <bios.h>
#include <gf.h>
#include <asiports.h>
void com1op()
{
    int stat;
    stat =

asiopen(COM1,(ASIN|BINARY|NORMALRX),4000,4000,120
0,P_ODD,2,8,ON,OFF);

    if(stat != ASSUCCESS)
    {
      printf("Error initializing Port COM1 - Status = %x\n",
           stat);
    }
}
```

function: tcode.c

```
/***************************************************
funtion tcode.c

Function to retrieve time code information from Datum time
code generator through the PC serial port.  Use com1open.c first
to open com port.

modified from Jessup [1990]

***************************************************/
#include <stdio.h>
#include <bios.h>
#include <gf.h>
#include <asiports.h>
void tcode(time)
char *time;
{
    int stat;
    int data1[7];
    int j,i=0;
/*  stat = asdtr(COM1,ON); */
    stat = asrts(COM1,ON);
    if(stat != ASSUCCESS)
    {
      printf("Error Setting RTS in COM1 - Status = %x\n",
           stat);
    }
    while(iscts(COM1,DIRECT))
    {
      if(!isrxempty(COM1))
      {
        data1[i++] = asigetc(COM1);
      }
    }
    stat = asrts(COM1,OFF);
/*  stat = asdtr(COM1,OFF); */
    data1[4]   = data1[4] & 0x3F;
    *time      = (char) ((data1[6] & 0x0F) + 0x30);
    *(time+1)  = (char) (((data1[5] & 0xF0) >> 4) + 0x30);
    *(time+2)  = (char) ((data1[5] & 0x0F) + 0x30);
    *(time+4)  = (char) (((data1[4] & 0xF0) >> 4) + 0x30);
    *(time+5)  = (char) ((data1[4] & 0x0F) + 0x30);
    *(time+7)  = (char) (((data1[3] & 0xF0) >> 4) + 0x30);
    *(time+8)  = (char) ((data1[3] & 0x0F) + 0x30);
    *(time+10) = (char) (((data1[2] & 0xF0) >> 4) + 0x30);
    *(time+11) = (char) ((data1[2] & 0x0F) + 0x30);
    *(time+13) = (char) ((data1[1] & 0x0F) + 0x30);
    *(time+14) = (char) (((data1[0] & 0xF0) >> 4) + 0x30);
    *(time+15) = (char) ((data1[0] & 0x0F) + 0x30);
}
```

Appendix I

Computer programs used in section II

This appendix contains the computer programs used in section II which includes §6, and §7. Programs running on personal computers were written in Microsoft C V6.0 or Microsoft Fortran V4.1. Programs running on the Spectrum TMS320C30 DSP board were written in TI C V4.1 and TI Assembler V4.1. The "make file" containing the compiler and linker directives is given in the comment section at the beginning of each program. Third party library drivers are also described in the comment section. Table I.1 gives an overview of the programs treated in this appendix. Figure I.1 shows a bloc diagram of the programs used in the measurement of sound-speed in §6 and §7.

program name	§	description
ladder.c	7	Acquires and processes acoustic pulses from field experiment. Extracts time-delay.
ladder.num	7	File containing the "filenumbers". The letters are a=5kHz, b=7.5kHz and so on. f=composite pulse.
ladder.nor	7	File containing the normalization for the time-delay. Multiply by 2 to get microseconds.
filter.c	7	Sets the cutoff frequencies for the high-pass and low-pass filter banks by activating two 8bits digital ports.
source.c	7	Selects the transmit hydrophone by activating an 8bits digital port.
pulse.c	7	Sets the pulse and signal generator.
pulse.ini	7	File containing parameters for pulse.c
tc.c	7	Function of ladder.c - reads Horita time code
ctemp.for	7	Function of ladder.c - reads Sea Bird SBE-3 temp. probe
tf.c	7	Cross-correlation and peak detection code for TMS320C30 DSP board.
tf.cmd	7	Command file for linking tf.c with TI C comiler V4.1
i.c	7	Ensemble of small functions used by tf.c and profilt.c
fft_rl.asm	7	Function of tf.c - forward FFT routine written in TI assembler
ifft_rl.asm	7	Function of tf.c - Inverse FFT routine written in TI assembler
s1024.asm	7	Function of fft_rl.asm and ifft_rl.asm - sine table
profilt.c	7	Processes raw pulse data. Performs band-pass filtering before cross-correlation. Computes attenuation.

Table I.1: Computer programs treated in this appendix.

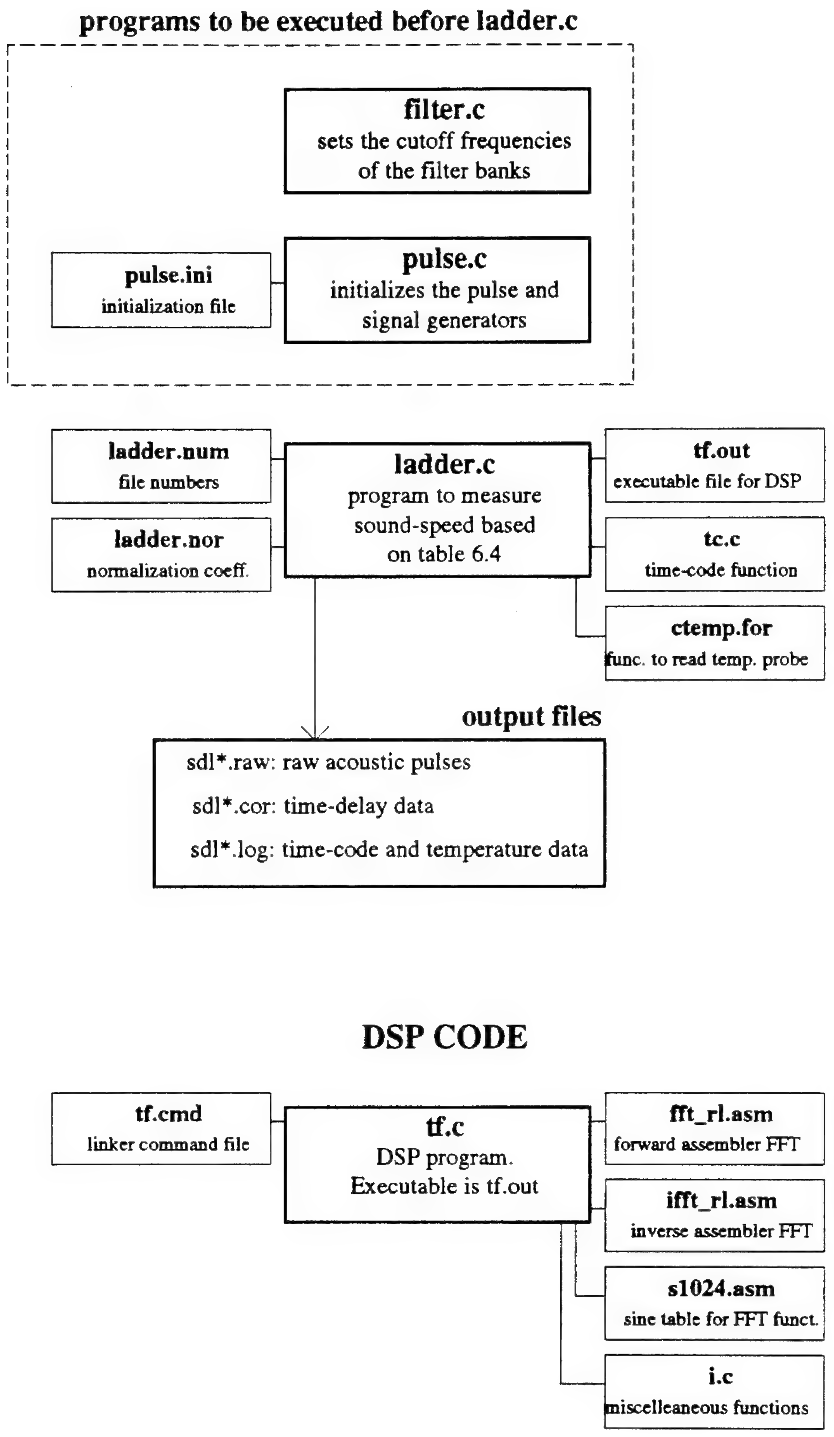

programs to be executed before ladder.c
filter.c
sets the cutoff frequencies of the filter banks
pulse.ini
initialization file
pulse.c
initializes the pulse and signal generators
ladder.num
file numbers
ladder.nor
normalization coeff.
ladder.c
program to measure sound-speed based on table 6.4
tf.out
executable file for DSP
tc.c
time-code function
ctemp.for
func. to read temp. probe
output files
sdl*.raw: raw acoustic pulses
sdl*.cor: time-delay data
sdl*.log: time-code and temperature data
DSP CODE
tf.cmd
linker command file
tf.c
DSP program. Executable is tf.out
fft_rl.asm
forward assembler FFT
ifft_rl.asm
inverse assembler FFT
s1024.asm
sine table for FFT funct.
i.c
miscelleaneous functions

Figure I.1: Bloc diagram of programs organization for the data acquisition system used in §6 and §7 for the measurements of sound-speed. Bold boxes are programs, all other boxes are input files, output files or subroutines. The (*) refers to various file numbers.

program: ladder.c

```
/*****************************************************
Program: ladder.c

Program is written in Microsoft C v6.0

The program acquires data from the hydrophone
buoy system to measure sound speed.

synopsis:    ladder letter repetitions
    letter = a,b,c...etc defines the pulse mode
    repetitions = optional, number of repetitions

Special libraries:
    mc025.lib
    Real Time Graphics library from Quinn-Curtis

    ctmper.lib
    library from Metrabyte to drive CTM-PER board

Special functions:
    lm30app.obj High-level language library for the
    TMS320C30 board. From Spectrum and LSI Ltd.

    lmcload.obj same as above

    tf.out Executable code written in
    TI C compiler (source is tf.c).

Typical batch file used to call ladder.exe: "ladder c 320"
The "c" means that the program will call ladderc.bat
which contains:
    echo off
    c:\pio\filter 1000 13200 0
    c:\pio\source 1 0 0 0 0 0 0 0
    c:\signa\pulse 24 10000 2.0 308

First written by: Eric Lamarre (08-21-92)

COMPILING AND LINKING DIRECTIVES:
ladder.exe: ladder.obj lm30app.obj lmcload.obj ctemp.obj tc.obj
LINK $**, $@, NUL, llibc7+llibfor7+mc025+graphics+ctmper
/NOD /NOE;

ladder.obj: ladder.c
CL /c /FPi87 /AL /G2 /Fs ladder.c
*****************************************************/

#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <process.h>
#include <stdlib.h>
#include <math.h>
#include <graph.h>
#include <time.h>
#include <dos.h>

#include "c:\c30tools\tms30.h"
#include "c:\egaa\isc16def.h"
#include "c:\egaa\isc16cmd.h"
#include "c:\mc025\rtstdhdr.h"
#include "c:\mc025\rtgsubs.h"
#include "c:\mc025\rtgraph.h"

#define SETUP_FILE_DIR "c:\\egaa\\"
#define SETUP_FILE_NAME "ladder"
#define NUM_OPTIONS 13

#define DSP_BOARDADR 0x290
#define DSP_PROGRAM_NAME "tf.out"
#define NUM_LOC 1
#define COMM0 0x30000
#define PCPROCEEDFLAG COMM0 + 0
#define DSPPROCEEDFLAG COMM0 + 1
#define MAX_LOCADR COMM0 + 2
#define ANSADR COMM0 + 3
#define DATAR1ADR COMM0 + 4
#define DATAR2ADR COMM0 + 5
#define MAXCOUNT 2000000

#define RC_BASE_ADDR 0x0300
#define SAMPLE_PER_BURST 0x0002
#define DELTAT 0x0002
#define POST_TRIGGER_DELAY 0x0700
#define N 1024
#define REPETITION 320
#define NUM_MODULE 6

#define FILENUM "c:\\egaa\\ladder.num"
#define NORM_COEFF "c:\\egaa\\ladder.nor"
#define RAM_FILE_DIR "f:\\"
#define EXP_NAME "sd"
#define CONFIG_NAME "l"
#define RAW_DATA_FILE_EXT ".raw"
#define COR_DATA_FILE_EXT ".cor"
#define COR_FILE_DIR "c:\\data\\"
#define LOG_FILE_DIR "c:\\data\\"
#define LOG_FILE_EXT ".log"

#define SYST_COM1 "copy f:\\*.raw c:\\data"
#define SYST_COM2 "del f:\\*.raw"

#define TIMEINT 320.0
#define SAMPLEINT 1.0
#define MINY 0.7
#define MAXY 1.3
#define RT 0.01
#define STPNT 1000.0

#define OFFSET 4
#define MESH 20

#define POWER_MULT_BASE_ADR 0x280

extern void _fortran ctemp(float _near * sal, float _near * depth,
                float _near * temp, float _near * soundsp);

void read_time_code(int *hr, int *min, int *sec, int *frame);
void EoLine(FILE *fp) { while(fgetc(fp) != '\n'); }
void initialize_boards(char option[1]);
void initialize_dsp(void);
void get_file_name(char option[1], char filename[30]);
void initialize_rc_board(char option[1], int module_num);
void sample_data(void);
void initialize_graph(realtype timeint);
```

```
rtstatpntr rtstat[NUM_MODULE];
int data1[N], data2[N];
float datar1[N], datar2[N], ans[2*N], max_loc[NUM_LOC],
sans[N];
float fmax_corr_loc[REPETITION*NUM_MODULE];

void main(int argc, char *argv[])
{
  FILE *fp;
  time_t t0_time, t1_time;
  realtype yvalues[1], timeint;
  int st, rep;
  int r, m, mm, i, j;
  int transmit_loc[] = { 1, 2, 4, 8, 16, 32, 1 };
  int hr1, min1, sec1, frame1, hr2, min2, sec2, frame2;
  float norm_coeff[NUM_MODULE];
  float sal = 35.0, depth = 0.5;
  float temp, soundsp;
  double intnum;
  unsigned long datar1adrloc, datar2adrloc, ansadrloc,
max_locadrloc;
  char option[1], filename[30], ram_file[30], cor_file[30],
log_file[30];

/* EXTRACT COMMAND LINE OPTION */

  if (argc < 2) { printf("Need at least one argument!\n"); exit(st);
}
  else strcpy(option,argv[1]);
  if (argc == 3) { rep = (int) atof(argv[2]), timeint = (realtype)
rep; }
  else { rep = REPETITION; timeint = (realtype) TIMEINT; }

/* OPEN FILE CONTAINING NORMALIZATION
COEFFICIENTS AND READ THEM */

  if ((fp = fopen(NORM_COEFF,"rt")) == NULL)
  { printf("Cannot open file %s\n",NORM_COEFF); exit(st); }
  for (i=0; i<NUM_MODULE; i++)
  { fscanf(fp,"%f\n",&norm_coeff[i]); EoLine(fp); }
  close(fp);

/* INITIALIZATION AND OPEN RAW DATA FILE */

  initialize_boards(option);
  outp(POWER_MULT_BASE_ADR+1, transmit_loc[0]);
  initialize_dsp();
    ansadrloc = Get32Bit(ANSADR,DUAL);
    datar1adrloc = Get32Bit(DATAR1ADR,DUAL);
    datar2adrloc = Get32Bit(DATAR2ADR,DUAL);
    max_locadrloc = Get32Bit(MAX_LOCADR,DUAL);
  get_file_name(option,filename);
  strcpy(ram_file,RAM_FILE_DIR);
  strcat(ram_file,filename);
  strcat(ram_file,RAW_DATA_FILE_EXT);
  if ((fp = fopen(ram_file,"wb")) == NULL)
  { printf("Cannot open file %s\n",ram_file); exit(st); }
  initialize_graph(timeint);

/* START MAIN REPETITION LOOP */

  read_time_code(&hr1, &min1, &sec1, &frame1);
  time(&t0_time);
  for (r=0; r<rep; r++)
  {
   for (m=0; m<NUM_MODULE; m++)
   {
    if (m==0) mm=5; else mm=m-1;
    initialize_rc_board(option,m);
    sample_data();
    DISABLE_ISC16(RC_BASE_ADDR);
/*
    for (i=0; i<N; i++) { data1[i] = 2047; data2[i] = 2047; }
    data1[0] = 2253; data2[136] = 2253;
    data1[1] = 2458; data2[137] = 2458;
    data1[2] = 2662; data2[138] = 2662;
    data1[3] = 2867; data2[139] = 2867;
    data1[4] = 3072; data2[140] = 3072;
    data1[5] = 2867; data2[141] = 2867;
    data1[6] = 2662; data2[142] = 2662;
    data1[7] = 2458; data2[143] = 2458;
    data1[8] = 2253; data2[144] = 2253;
*/
/* SWITCH CHANNEL ON POWER MULTIPLEXER */

    outp(POWER_MULT_BASE_ADR+1,
transmit_loc[m+1]);

/* WRITE DATA TO DISK */

    if (modf( (double)r/2.0, &intnum ) > 0.1)
    {
      fwrite(data1, sizeof(int), N, fp);
      fwrite(data2, sizeof(int), N, fp);
    }

/* COMPUTE CROSS-CORRELATION */

    for (i=0; i<N; i++) { datar1[i]=data1[i]; datar2[i]=data2[i];
}
    if (m==0) for (i=600; i<N; i++) datar2[i] = 2047.0;
    WrBlkFlt(datar1adrloc,DUAL,N,datar1);
    WrBlkFlt(datar2adrloc,DUAL,N,datar2);
    Put32Bit(DSPPROCEEDFLAG,DUAL,0x1L);
    if (r!=0 || m!=0)
    {
      while(Get32Bit(PCPROCEEDFLAG,DUAL) != 0x1L);
      Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
      RdBlkFlt(max_locadrloc,DUAL,NUM_LOC,max_loc);
      Put32Bit(DSPPROCEEDFLAG,DUAL,0x1L);
      fmax_corr_loc[r*NUM_MODULE+m-1] = max_loc[0];

/* UPDATE GRAPHICS HERE */

      yvalues[0] = (realtype) (max_loc[0] / norm_coeff[mm]);
      rtupdatescrollgraph(rtstat[mm], yvalues);

    }
   }
  }

/* READ DATA HERE FOR THE LAST CROSS-
CORRELATION */

  while(Get32Bit(PCPROCEEDFLAG,DUAL) != 0x1L);
  Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
  RdBlkFlt(max_locadrloc,DUAL,NUM_LOC,max_loc);
  Put32Bit(DSPPROCEEDFLAG,DUAL,0x1L);
  fmax_corr_loc[rep*NUM_MODULE-1] = max_loc[0];
```

```
/* UPDATE GRAPHICS HERE FOR THE LAST POINT */

  yvalues[0] = (realtype) (max_loc[0] /
norm_coeff[NUM_MODULE-1]);
  rtupdatescrollgraph(rtstat[NUM_MODULE-1], yvalues);

  time(&t1_time);
  read_time_code(&hr2, &min2, &sec2, &frame2);
  close(fp);

/* SAVE CROSS-CORRELATION RESULTS TO DISK */

  strcpy(cor_file,COR_FILE_DIR);
  strcat(cor_file,filename);
  strcat(cor_file,COR_DATA_FILE_EXT);
  if ((fp = fopen(cor_file,"wt")) == NULL)
  { printf("Cannot open file %s\n",cor_file); exit(st); }
  for (r=0; r<rep; r++)
  {
    for (m=0; m<NUM_MODULE; m++)
      fprintf(fp,"%8.2f
",fmax_corr_loc[r*NUM_MODULE+m]);
    fprintf(fp,"\n");
  }
  close(fp);

/* TRANSFER FILES FROM RAM DISK TO OPTICAL DISK
*/

  system(SYST_COM1); system(SYST_COM2);

/* TAKE TEMPERATURE AND WRITE LOG FILE TO DISK
*/

  ctemp(&sal,&depth,&temp,&soundsp);
  strcpy(log_file,LOG_FILE_DIR);
  strcat(log_file,filename);
  strcat(log_file,LOG_FILE_EXT);
  if ((fp = fopen(log_file,"wt")) == NULL)
  { printf("Cannot open file %s\n",log_file); exit(st); }
  fprintf(fp,"%ld %ld %.1lf %s", t0_time, t1_time,
difftime(t1_time,t0_time),
                    asctime(localtime(&t0_time)));
  fprintf(fp,"%.4f  %.4f  %.4f  %.4f\n", sal, depth, temp,
soundsp);
  fprintf(fp,"%02X:%02X:%02X:%02X\n",hr1, min1, sec1,
frame1);
  fprintf(fp,"%02X:%02X:%02X:%02X\n",hr2, min2, sec2,
frame2);
  close(fp);

/* CLOSE GRAPHICS WINDOW */

/* while(!kbhit()); */
  rtclosegraphics(rtstat,1);
  printf("%s\n",filename);
  printf("%ld %ld %.1lf %s", t0_time, t1_time,
difftime(t1_time,t0_time),
                    asctime(localtime(&t0_time)));
  printf("%.4f  %.4f  %.4f  %.4f\n", sal, depth, temp, soundsp);
  printf("%02X:%02X:%02X:%02X\n",hr1, min1, sec1,
frame1);
  printf("%02X:%02X:%02X:%02X\n",hr2, min2, sec2,
frame2);
```

```
} /* END OF MAIN */


void initialize_boards(char option[1])
{
  FILE *fp;
  int i, st, er;
  char *setup_file;

  strcpy(setup_file,SETUP_FILE_DIR);
  strcat(setup_file,SETUP_FILE_NAME);
  strcat(setup_file,option);
  system(setup_file);
}

void initialize_dsp(void)
{
  unsigned short loadstat;
  long i;

/* DOWNLOAD DSP CODE - CODE MUST BE IN SAME DIR
AS MAIN PROG. */

  SelectBoard(DSP_BOARDADR);
  loadstat = coffLoad(DSP_PROGRAM_NAME);
  if (loadstat != 0)
  {
    printf("\n\nError During Program Load.\n");
    printf("coffLoad() returned %x\n\n", loadstat);
    exit (0);
  }
  Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
  Put32Bit(DSPPROCEEDFLAG,DUAL,0x0L);
  Reset();

/* WAIT FOR DSP BOARD TO INITIALIZE */

  for (i=0; i<MAXCOUNT &&
(Get32Bit(PCPROCEEDFLAG,DUAL) != 0x1L); i++);
  if (i==MAXCOUNT) { printf("Timeout waiting for DSP.\n");
exit(0); }
  Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
}

void get_file_name(char option[1], char filename[30])
{
  FILE *fp;
  int i, st, num;
  char op[NUM_OPTIONS][2], numfile[NUM_OPTIONS][6],
*padded_num;

  if ((fp = fopen(FILENUM,"rt")) == NULL)
  { printf("Cannot open file %s\n",FILENUM); exit(st); }
  for (i=0; i<NUM_OPTIONS; i++)
  {
    fscanf(fp,"%s %s",op[i],numfile[i]); EoLine(fp);
  }
  close(fp);

  for (i=0; i<NUM_OPTIONS; i++)
  {
    if (op[i][0] == option[0])
    {
      num = atoi(numfile[i]) + 1;
      sprintf(numfile[i],"%d",num);
      break;
    }
    else num = 0;
  }

  if ((fp = fopen(FILENUM,"wt")) == NULL)
  { printf("Cannot open file %s\n",FILENUM); exit(st); }
  for (i=0; i<NUM_OPTIONS; i++) fprintf(fp,"%s
%s\n",op[i],numfile[i]);
  close(fp);

  sprintf(padded_num,"%04d",num);
  strcpy(filename,EXP_NAME);
  strcat(filename,CONFIG_NAME);
  strcat(filename,option);
  strcat(filename,padded_num);
}

void initialize_rc_board(char option[1], int module_num)
{
  unsigned char channel[16];
  int i, offset;

/* ASSIGN PROPER CHANNEL TO BE SAMPLED BASED
ON option AND module_num */

  switch (option[0])
  {
    case 'a':case 'b':case 'c':case 'd':case 'e':case 'f':case 'g':case 'h':
      offset = 0; break;
    case 'i':case 'j':case 'k':case 'l':case 'm':
      offset = 8; break;
    default:
      offset = 0; break;
  }

  for(i=0; i<16; i+=2) channel[i] = 7 + offset;
  for(i=1; i<16; i+=2) channel[i] = module_num + offset;

/* INITIALIZE */

  DISABLE_ISC16(RC_BASE_ADDR);

SELECT_INTERNAL_SAMPLE_CLOCK(RC_BASE_ADDR);
  SET_MUX_RAM(channel, RC_BASE_ADDR);
  SET_TRIGGER_CONTROL(EXTERNAL_TRIGGER,
POSITIVE_POLARITY,
                TRIGGER_SLOPE, RC_BASE_ADDR);
  SET_TRIGGER_CHANNEL(CHANNEL_A,
IRQ3_POST_TRIGGER, D2_D1_00, RC_BASE_ADDR);
  SET_INTERNAL_SAMPLE_RATE(DELTAT,
RC_BASE_ADDR);
  SET_SAMPLE_COUNT(SAMPLE_PER_BURST,
RC_BASE_ADDR);
  SET_POST_TRIGGER_DELAY(POST_TRIGGER_DELAY,
RC_BASE_ADDR);
}

void sample_data(void)
{
  FILE *fp;
  int i, k;

/* SET MEMORY BANKS AND START DATA
ACQUISITION */


SELECT_BANK_B_MANUAL_SWITCHING(RC_BASE_AD
DR);
  RESET_BANK_A_POINTER_AND_SIZE(BUFFER_2K,
RC_BASE_ADDR);
  ENABLE_ACQUISITION(RC_BASE_ADDR);
  for (i = 0; i < 400; i++) { }
  ENABLE_TRIGGER(RC_BASE_ADDR);
  WAIT_FOR_POST_TRIGGER_DELAY(RC_BASE_ADDR);

SELECT_BANK_A_MANUAL_SWITCHING(RC_BASE_AD
DR);

/* READ DATA FROM BANK A AND STORE IN ARRAYS */

  for (i=0; i<N; i++)
  {
    READ_MEMORY(data1[i]);
    for (k=0; k<1; k++);
    READ_MEMORY(data2[i]);
    for (k=0; k<1; k++);
  }
}

void initialize_graph(realtype timeint)
{
  realtype sampleint=SAMPLEINT;
  realtype miny=MINY, maxy=MAXY;
  realtype rt=RT;
  realtype lalarm=2*MINY, halarm=2*MAXY, stpnt=STPNT;
  tagtype tags[]={' '};
  char title[40], units[20]="TIME DELAY uS", tch[5];
  int lc[]={14}, lf[]={0};
  int nt=1, grid=0, ratchf=0;
  int i;

/* INITIALIZE THE GRAPHICS ADAPTER, SET UP REAL
TIME WINDOWS*/

  rtinitgraphics(-2,defaultfontdir,6,rtstat,1);
  rtsetpercentwindow(rtstat[0],0.01,0.01,0.49,0.33);
  rtsetpercentwindow(rtstat[1],0.01,0.34,0.49,0.66);
  rtsetpercentwindow(rtstat[2],0.01,0.67,0.49,0.99);
  rtsetpercentwindow(rtstat[3],0.50,0.01,0.99,0.33);
  rtsetpercentwindow(rtstat[4],0.50,0.34,0.99,0.66);
  rtsetpercentwindow(rtstat[5],0.50,0.67,0.99,0.99);

/* START LOOP WHICH SETS UP THE SCROLLING
GRAPHS */

  for (i=0; i<6; i++)
  {
    strcpy(title, "MODULE ");
    sprintf(tch , "%d", i);
    strcat(title, tch);
    rtinitwindowcolors(rtstat[i],0,0,1,4,15,15,15);
    rtsetupscrollgraph(rtstat[i],timeint, sampleint, miny, maxy, rt,
            nt, grid, lalarm, halarm, stpnt, 1,2,title, units,
            tags, lc, lf, ratchf);
    rtborderwindow(rtstat[i],15);
  }
}
```

input file: ladder.num

```
a 300
b 300
c 300
d 300
e 300
f 300
g 300
h 300
i 300
j 300
k 300
l 300
m 300
```

input file: ladder.nor

```
139.0      normalization coefficient for module 0
139.0      normalization coefficient for module 1
139.0      normalization coefficient for module 2
139.0      normalization coefficient for module 3
139.0      normalization coefficient for module 4
139.0      normalization coefficient for module 5
```

program: filter.c

```
/***************************************************
Program: filter.c

Program is written in Microsoft C v6.0

Program to set the cutoff frequencies on the filter banks.  The
PIO-12 or PIO-24 card by Metrabyte can be used. It is assumed
that the PA port is used for the high-pass and the PB port for the
low-pass. The program also sets the PC port.

synopsis: at the dos prompt type: filter HP LP PC
    where HP: high-pass Fc to lowest multiple of 10
    LP: low-pass Fc to lowest multiple of 200
    PC: any value between 0 and 255

    The high-pass formula is: Fc = 10 * (1 +PAbyte)
    The low-pass formula is:  Fc = 200 * (1 + PBbyte)


First written by: Eric Lamarre (07-27-92)

Compiling and linking commands:
filter.exe: filter.obj
LINK $**, $@, NUL, llibc7 /NOD /NOE;

filter.obj: filter.c
CL /c /FPi87 /AL /G2 /Fs filter.c
***************************************************/

#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

#define BASE_ADR 0x284
#define MIN_FC_HP 10.0
#define MIN_FC_LP 200.0
#define PC_CONST 1.0

void main(int argc, char *argv[ ])
{
  unsigned base;
  int PAbyte, PBbyte, PCbyte;
  int st;

/* IDENTIFY COMMANDS PASSED AT THE COMMAND
PROMPT */

  if (argc != 4)
  {
   printf("Wrong # of arguments. Usage: filter PA PB PC\n");
   exit(st);
  }
  else
  {
   PAbyte = (int) (atof(argv[1]) / MIN_FC_HP) - 1;
   PBbyte = (int) (atof(argv[2]) / MIN_FC_LP)- 1;
   PCbyte = (int) (atof(argv[3]) / PC_CONST);
  }

  if (PAbyte<0 || PAbyte>255) printf("1st arg out of 10-2560
range\n");
  if (PBbyte<0 || PBbyte>255) printf("2nd arg out of 200-51200
range\n");
  if (PCbyte<0 || PCbyte>255) printf("3rd argument out of 0-255
range\n");

/* INITIALIZE PIO-12 OR PIO-24 BOARD */

  base = BASE_ADR;
  outp(base+3,0x80); /* 0x80 = 1000 000 */

/* SEND DATA TO: PA PORT (base+0), PB PORT (base+1)
and PC PORT (base+2) */

  outp(base+0,PAbyte);
  outp(base+1,PBbyte);
  outp(base+2,PCbyte);

/* REMOVE COMMENTS IF YOU WISH TO PRINT THE
CUTOFFS FREQUENCIES */

  printf("filter HP=%.0fHz  LP=%.0fHz
PCport=%d\n",MIN_FC_HP * (1+PAbyte),
        MIN_FC_LP*(1+PBbyte), PCbyte);
}
```

program: source.c

```
/***************************************************
Program: source.c

Program is written in Microsoft C v6.0

Program to turn on/off the electro-mechanical relays (8) of the
ERA-01 Metrabyte module to be driven by PIO-12 Metrabyte
PC card. The PB port of the PIO-12 controls the ERA-01.

synopsis: at the dos prompt type: source ch0 ch1 ... ch7
    where ch0, ch1, ch2 ... ch7 are the channels
    you wish to turn on with a 1 or off with a 0.
    ex: source 1 0 0 0 0 0 0 0 turns on ch0.

    All relays are off with PBbyte = 0.
    Relay 0 is turned on by PBbyte = 1.
    All relays are on with PBbyte = 255.

First written by: Eric Lamarre (07-27-92)

Compiling and linking commands:
source.exe: source.obj
LINK $**, $@, NUL, llibc7 /NOD /NOE;
source.obj: source.c
CL /c /FPi87 /AL /G2 /Fs source.c
**************************************************/

#include <stdio.h>
#include <stdlib.h>
#include <conio.h>

#define BASE_ADR 0x280

/* MAIN PROGRAM */

void main(int argc, char *argv[ ])
{
  unsigned base;
  int PBbyte, st, i;

/* IDENTIFY COMMANDS PASSED AT THE COMMAND
PROMPT */

  if (argc != 9)
  {
   printf("Wrong # of arguments\n");
   printf("Usage: source ch0 ch1 ch2 ch3 ch4 ch5 ch6 ch7\n");
  }
  else
  {
   for (i=1; i<9; i++)
   {
    if (atoi(argv[i]) != 0 && atoi(argv[i]) != 1)
    {
      printf("Argument %d not 0 (off) or 1 (on)\n",i);
      exit(st);
    }
   }
   PBbyte = 1*atoi(argv[1]) + 2*atoi(argv[2]) + 4*atoi(argv[3])
          + 8*atoi(argv[4]) + 16*atoi(argv[5]) + 32*atoi(argv[6])
          + 64*atoi(argv[7]) + 128*atoi(argv[8]);
  }

/* INITIALIZE PIO-12 BOARD */

  base = BASE_ADR;
  outp(base+3,0x80); /* 0x80 = 1000 000 */

/* SEND DATA TO: PB PORT AT (base+1) */

  outp(base+1,PBbyte);

/* REMOVE COMMENTS IF YOU WISH TO PRINT PBbyte
*/

  printf("source %d %d %d %d %d %d %d %d\n",atoi(argv[1]),
atoi(argv[2]),
       atoi(argv[3]), atoi(argv[4]), atoi(argv[5]), atoi(argv[6]),
       atoi(argv[7]), atoi(argv[8]), atoi(argv[9]) );

}
```

program: pulse.c

```
/**************************************************
Program: pulse.c

Modification of func_tst.c found in Signametrics SM20 manual

Program is written in Microsoft C v6.0
Uses Signametrics drivers.h and drivers.lib

The program operates the SM1020 signal generator and the
SM1005 pulse generator board. Uses initialization file passed at
command line for parameter setup.

synopsis: pulse p_rep f_out amp #wave
    where: p_rate = pulse repetition rate
    f_out = desired output frequency
    amp = amplitude of signal in Volts
    #wave = file number ('pulse'###'.wave')

The program also reads the file 'pulse.ini' which contains several
parameter settings.

First written by: Eric Lamarre (06-16-92)
modified (07-27-92)

Compiling and linking commands:
pulse.exe: pulse.obj
LINK $**, $@, NUL, llibc7+drivers /NOD /NOE;

pulse.obj: pulse.c
CL /c /FPi87 /AL /G2 /Fs pulse.c
**************************************************/

#include <stdio.h>
#include <string.h>
#include <math.h>
#include "drivers.h"

/* FORMAL FUNCTION DECLARATIONS */

extern void set_level(double level, struct state *stat_ptr);
extern void set_offset(double offst, struct state *stat_ptr);
extern int  tab_load( FILE *fp, struct state *stat_ptr);
extern int  suspend( struct state *stat_ptr );
extern int  resume(struct state *stat_ptr );
extern void set_phase( int phas, struct state *stat_ptr );
extern unsigned calc_phase(int phas, struct state *stat_ptr);
extern void set_freq(double freq ,struct state *stat_ptr );
extern void  set_pulse(double p_rate, double p_wid, struct state
*stat_ptr);
extern void sel_sharp( struct state *stat_ptr);
extern void sel_lpf( struct state *stat_ptr);
extern int  sel_trig(char t_source, struct state *stat_ptr);
extern void inv_trig( struct state *stat_ptr );
extern void uninv_trig( struct state *stat_ptr );
extern void sel_pls( struct state *stat_ptr );
extern void sel_func( struct state *stat_ptr );
extern int  sel_analog(char a_mode, struct state *stat_ptr);
extern int  sel_clock(char c, struct state *stat_ptr);
extern int  rel_phase( struct state *A, struct state *B );
extern void un_rel_phase( struct state *A, struct state *B );
extern int  sel_mode(char mode, struct state *stat_ptr);
extern void s_trg_h( struct state *stat_ptr);
extern void s_trg_l( struct state *stat_ptr);
extern void burst(unsigned bcnt, struct state *stat_ptr);
extern void triang_wav(struct state *stat_ptr);    /* void */
extern void havertri_wav(struct state *stat_ptr);
extern void trapez_wav(struct state *stat_ptr);
extern void noise_wav(struct state *stat_ptr);
extern void squar_wav(struct state *stat_ptr);
extern void dc(double dc, struct state *stat_ptr);
extern void pos_ramp_wav( struct state *stat_ptr);
extern void neg_ramp_wav( struct state *stat_ptr);
extern void zero_wav( struct state *stat_ptr);
extern void pfs_wav( struct state *stat_ptr);
extern void nfs_wav( struct state *stat_ptr);
extern void init( struct state *stat_ptr );
extern void la_strob(int data, int la, struct state *stat_ptr);
extern void cohere_freq(struct state *A, struct state *B);
extern void tab_preload( struct state *stat_ptr );
extern void step_wav( int phase_step, struct state *stat_ptr);

/* GLOBAL VARIABLES */

char system_mode='Y'; /* current dual board mode */
char current_fpg;
float fscaler = 1.0f; /* default clock prescaler, 1 for 10MHz/4
factor */

struct state fpg_a =    /* first card structure (excluding wave
image) */
{
  0x87,0,0,0,0,253,0,0,
  127,255,0x1c,0,0,0x01,0,0,
  0,
  0x340,
  8,
  1,
  'N',
  'N',
  0,
  1e3,
  1e3,
  2e-4,
  0.0,
  0.0,
  4350,
  'N'
};

struct state fpg_b =    /* second card vital data store (optional)
*/
{
  0x87,0,0,0,0,253,0,0,
  127,255,0x1c,0,0,0x01,0,0,
  0,
  0x348,
  8,
  1,
  'N',
  'N',
  0,
  1e3,
  1e3,
  1.0e-4,
  5.0,
  0.0,
  4350,
  'Y'
};

void EoLine(FILE *fp) { while(fgetc(fp) != '\n'); }

#define N 8192
#define FILE_DIR "c:\\signa\\arbwave\\"
#define FILE_NAME "pulse"
#define FILE_EXT ".wav"
#define PULSE_INI "c:\\signa\\pulse.ini"

/* MAIN PROGRAM */

void main(int argc, char *argv[ ])
{
  FILE *fp, *fopen();
  int st;
  int phas, bcnt, num_pt_per;
  double amp, offst, freq, f_out;
  double p_rate, p_wid;
  char char_mode, char_trig;
  char filetab[30], *filenum;
  struct state *pointer;

/* INITIALIZE POINTERS */

  pointer = &fpg_a;
  current_fpg = 'A';
  init(pointer);
  pointer = &fpg_b;
  current_fpg = 'B';
  init(pointer);

/* INPUT STANDARD PARAMETERS FROM pulse.ini */

  if( (fp=fopen(PULSE_INI,"r")) != NULL )
  {
    fscanf(fp,"%lf\n",&p_rate); EoLine(fp);
    fscanf(fp,"%lf\n",&p_wid); EoLine(fp);
    fscanf(fp,"%lf\n",&f_out); EoLine(fp);
    fscanf(fp,"%d\n",&num_pt_per); EoLine(fp);
    fscanf(fp,"%lf\n",&amp); EoLine(fp);
    fscanf(fp,"%lf\n",&offst); EoLine(fp);
    fscanf(fp,"%d\n",&phas); EoLine(fp);
    fscanf(fp,"%1c\n",&char_mode); EoLine(fp);
    fscanf(fp,"%1c\n",&char_trig); EoLine(fp);
    fscanf(fp,"%d\n",&bcnt); EoLine(fp);
    fscanf(fp,"%s\n",filenum);
  }
  else printf("Error, can't open pulse.ini file.\n");
  fclose(fp);

/* IDENTIFY OTHER COMMANDS PASSED TO PROGRAM
- MODIFY PARAMS ACCORDINGLY */

  if (argc == 5)
  {
    p_rate = atof(argv[1]);
    f_out = atof(argv[2]);
    amp = atof(argv[3]);
```

```
   filenum = argv[4];
  }
  else if (argc != 1) { printf("Need 4 arguments or none!");
exit(st); }

/* COMPUTATION OF FREQUENCY TO SPECIFY TO
BOARD */

  freq = (num_pt_per * f_out) / N;
  if (freq<=p_rate) {printf("\nPulse rep. rate too high!\n");
exit(st);}
  if (p_wid >= 1.0/p_rate) { printf("\nPulse width too wide!\n");
exit(st); }

/* LOAD DATA TABLE */

  strcpy(filetab,FILE_DIR);
  strcat(filetab,FILE_NAME);
  strcat(filetab,filenum);
  strcat(filetab,FILE_EXT);
  if( (fp=fopen(filetab,"r")) != NULL ) tab_load( fp, pointer );
  else { printf("Error, opening file %s\n",filetab); exit(st); }
  fclose(fp);

/* SEND COMMANDS TO SM1005 BOARD */

  pointer = &fpg_a;
  current_fpg = 'A';
  sel_pls(pointer);
  set_pulse(p_rate,p_wid,pointer);

/* SEND COMMANDS TO SM1020 BOARD */

  pointer = &fpg_b;
  current_fpg = 'B';
  set_freq(freq,pointer);
  set_phase(phas,pointer);
  set_level(amp,pointer);
  set_offset(offst,pointer);
  sel_sharp(pointer);
  sel_mode(char_mode,pointer);
  sel_trig(char_trig,pointer);
  uninv_trig(pointer);
  burst(bcnt,pointer);

/* PRINT BRIEF MESSAGE */

  printf("pulse %.1fHz %.1fHz %.1fV %dfile\n",p_rate, f_out,
       amp, atoi(filenum));
}
```

input file pulse.ini

```
1000.0      pulse rate (Hz)
1.0e-4      pulse width (seconds)
20000.0         frequency at output (Hz)
512         number of data points per period (#)
6.0         amplitude peak to peak (volts)
0.0     DC offset (volts)
0       phase offset (1/10 degrees; must be int)
T       mode (F=free, T=trigger)
P       trigger mode (P=master pulse, X=ext)
1       burst count (#, must be int)
3       file # to retrieve waveform
```

function: tc.c

```
/***************************************************
Function: tc.c

Function written in Microsoft C v6.0

Function used to read the TRG-50PC time-code generator by
Horita. The ROM BIAS serial interface was taken from the book
SERIAL COMMUNICATIONS IN C AND C++ by Mark
Goodwin (chapter 3). Note: this routine was timed to take 33mS
to execute on an ALR 386 33MHz.

First written by: Eric Lamarre (October 1992)

Compiling directives:
tc.obj: tc.c
CL /c /FPi87 /AL /G2 /Fs tc.c
***************************************************/

#include <stdio.h>
#include <stdlib.h>
#include <dos.h>
#include <time.h>

#define PORT 1
#define BAUDRATE 9600

#define TRUE 1
#define FALSE 0
#define TIMEOUT 2.0

void open_port(int n, int b);
int get_serial(int n);
int in_ready(int n);

void read_time_code(int *hr, int *min, int *sec, int *frame)
{
  time_t t0, t1;
  unsigned int data[10], gi[4], st, i, j;
  int FFi;

/* OPEN SERIAL PORT */
  open_port(PORT, BAUDRATE);

/* TAKE 10 BYTES OUT OF SERIAL PORT. IF NO DATA
ON SERIAL PORT THEN LOAD 88 */
  for (i=0; i<10; i++)
  {
    time(&t0);
    while (!in_ready(PORT))
    {
      time(&t1);
      if (difftime(t1,t0) > TIMEOUT)
      {
        data[0] = 0xFF;
        for (j=1; j<10; j++) data[j] = 0x88;
        goto t_out;
      }
    }
    data[i] = get_serial(PORT) & 0xFF;
  }

/* THE FIRST 4 BYTES ABOVE 0xFF ARE
HOURS:MINUTES:SECS:FRAMES RESPECTIVELY */
/* THE DATA IS IN BINARY CODED DECIMAL FORMAT
SO PRINTING AS HEX WILL SHOW *
/* CORRECT TIME */
  t_out:
  for (i=0; i<10; i++) if (data[i] == 0xFF) FFi = i;
  if ((FFi - 1) < 0) gi[0] = 10+(FFi-1); else gi[0] = (unsigned int)
FFi-1;
  if ((FFi - 2) < 0) gi[1] = 10+(FFi-2); else gi[1] = (unsigned int)
FFi-2;
  if ((FFi - 3) < 0) gi[2] = 10+(FFi-3); else gi[2] = (unsigned int)
FFi-3;
  if ((FFi - 4) < 0) gi[3] = 10+(FFi-4); else gi[3] = (unsigned int)
FFi-4;

  *hr=data[gi[0]]; *min=data[gi[1]]; *sec=data[gi[2]];
*frame=data[gi[3]];
}

/* FUNTION open_port TO OPEN SERIAL PORT WITH ROM
BIOS ROUTINE */
void open_port(int n, int b)
{
  union REGS regs;

/* LOAD ah WITH THE FUNCTION CODE */
  regs.h.ah = 0x00;

/* LOAD al WITH THE BAUD RATE AND 8-N-1 */
  switch(b)
  {
    case 9600: regs.h.al = 0xe3; break;
    case 4800: regs.h.al = 0xc3; break;
    case 2400: regs.h.al = 0xa3; break;
    case 1200: regs.h.al = 0x83; break;
    case 300: regs.h.al = 0x43; break;
    case 150: regs.h.al = 0x23; break;
    default: regs.h.al = 0xe3;
  }
  if (n == 1) regs.x.dx = 0;
  else regs.x.dx = 1;

/* CALL THE ROM BIOS ROUTINE */
  int86(0x14, &regs, &regs);
}

/* FUNCT get_serial TO GET A BYTE FROM SERIAL PORT
USING ROM BIOS ROUTINE */
int get_serial(int n)
{
  union REGS regs;

/* LOAD ah WITH THE FUNCTION CODE */
  regs.h.ah = 0x02;

/* LOAD dx WITH THE PORT NUMBER */
  if (n == 1) regs.x.dx = 0;
  else regs.x.dx = 1;

/* CALL THE ROM BIOS ROUTINE */
  int86(0x14, &regs, &regs);

/* RETURN EOF IF TIMEOUT */
  if (regs.h.ah & 0x80) return EOF;
  return regs.h.al;
}

/* FUNCTION TO CHECK IF A BYTE IS PRESENT AT THE
SERIAL PORT */
int in_ready(int n)
{
  union REGS regs;

/* LOAD ah WITH FUNCTION MODE */
  regs.h.ah = 0x03;

/* LOAD dx WITH PORT NUMBER */
  if (n == 1) regs.x.dx = 0;
  else regs.x.dx = 1;

/* CALL THE ROM BIOS ROUTINE */
  int86(0x14, &regs, &regs);

/* CHECK FOR RECEIVE DATA READY */
  if (regs.h.ah & 1) return TRUE;
  return FALSE;
}
```

program: ctemp.for

```
$NOTRUNCATE
C
C SUBROUTINE CTEMP
C
C SUBROUTINE WRITTEN IN MICROSOFT FORTRAN V4.1
C
C SUBROUTINE WRITTEN FOR INTERFACING WITH
C MICROSOFT C V6.0
C
C SUBROUTINE TO SAMPLE 3 DATA POINTS ON
C METRABYTE CTM-PER
C PERIOD COUNTER AND RETURN THE 3RD ONE.
C
C THE CTM-PER PERIOD COUNTER IS USED
C TO EXTRACT THE TEMPERATURE
C INFORMATION FROM THE SINUSOIDAL OUTPUT
C OF THE TEMPERATURE PROBE
C
C 04-21-92: FIRST WRITTEN BY ERIC LAMARRE.
C
C COMPILING AND LINKING NOTES
C ctemp.obj: ctemp.for
C FL /c /4Ydb /4Nt /G2 /Fs ctemp.for
C
      subroutine ctemp(sal,depth,temp,soundsp)
C
      integer*4 address
      integer*2 base
      integer*4 delta(3)
      integer*2 dmalev
      integer*2 flag
      integer*4 i, j, k, l
      integer   ihr1, imin1, isec1, i100th1
      integer   ihr2, imin2, isec2, i100th2
      integer*2 intlev
      integer*2 mode0, mode1, mode2, mode8, mode9, mode10
      integer*2 off1, off2
      integer*2 param(15)
      integer*4 rawdata(4)
      integer*2 recycle
```

```
    integer*2 samples
    integer*2 seg1, seg2
    integer*4 t1_100th, t2_100th, tresh
C
    real*8 a, b, c, d, aa, bb, cc, dd
    real*8 divider
    real*8 fo, f, tk
    real*4 sal[NEAR], depth[NEAR], temp[NEAR],
soundsp[NEAR]
C
    data base/#360/
    data dmalev/1/
    data samples/4/
    data tresh/100/
    data intlev/3/
    data mode0/0/, mode1/1/, mode2/2/, mode8/8/, mode9/9/,
mode10/10/
    data recycle/0/
    data divider/256/
    data a/3.67568997e-3/, b/6.01392288e-4/
    data c/1.45949681e-5/, d/1.59732173e-6/
    data fo/6675.02/
C
C   INITIALIZE CTM-PER BOARD
C
    param(1) = base
    param(2) = intlev
    param(3) = dmalev
    call fctmper(mode0, param, flag)
    if(flag .ne. 0) print *,' Error mode0, flag = ',flag
C
C   SET GATING, EDGES AND SCALING OPTIONS
C
    do 10 i = 1, 5
      param(i) = 0
  10 continue
    call fctmper(mode2, param, flag)
    if(flag .ne. 0) print *,' Error mode2, flag = ',flag
C
C   SET DMA MODE
C
    address = locfar(rawdata(1))
    seg1 = address / #10000
    off1 = address - (seg1 * #10000)
    param(1) = samples
    param(2) = seg1
    param(3) = off1
    param(4) = recycle
    call gettim(ihr1, imin1, isec1, i100th1)
    t1_100th = ihr1*60*60*100 + imin1*60*100 + isec1*100 +
i100th1
    call fctmper(mode8, param, flag)
    if(flag .ne. 0) print *,' Error mode8, flag = ',flag
C
C   MONITOR END OF DMA
C
 999 call fctmper(mode10, param, flag)
    if(flag .ne. 0) print *,' Error mode10, flag = ',flag
    call gettim(ihr2, imin2, isec2, i100th2)
    t2_100th = ihr2*60*60*100 + imin2*60*100 + isec2*100 +
i100th2
    if (t2_100th-t1_100th .gt. tresh) goto 777
    if (param(3) .lt. samples) then
      go to 999
    endif
C
C   SHUT DOWN DMA
C
C   param(1) = 0
C   flag = 0
C   call fctmper(mode1, param, flag)
C   if(flag .ne. 0) print *,' Error mode1, flag = ',flag
C
C   CONVERT TO DELTAS
C
    do 20 i = 1, samples
      rawdata(i) = rawdata(i) / #10
  20 continue
C
    do 30 i = 2, samples
      if (rawdata(i) .gt. rawdata(i-1)) then
        delta(i-1) = rawdata(i) - rawdata(i-1)
      else
        delta(i-1) = (268435456 - rawdata(i-1)) + rawdata(i)
      endif
  30 continue
C
C   APLLY TRANSFER FUNCTION ON LAST delta TO
FIND TEMPERATURE
C
    f = (10.0e6 / delta(3)) * divider
    aa = a
    bb = b * (log(fo / f))
    cc = c * ((log(fo / f))**2)
    dd = d * ((log(fo / f))**3)
    tk = aa + bb + cc + dd
    temp = (1/tk) - 273.15
C
C   APLLY TRANSFER FUNCTION ON temp TO GET
soundsp
C
    soundsp = 1449.2 + 4.6*temp - 0.055*(temp**2) +
0.00029*(temp**3)
   1        + (1.34 - 0.010*temp)*(sal - 35) + 0.016*depth
    goto 666
 777 soundsp = 0.0
    temp = 0.0
C
 666 end
```

program: tf.c

```
/**************************************************
TF.C - DSP C program to compute cross-correlation, locate peak
of cross-correlation and interpolate data around peak. Program
uses a forward and inverse fft routine written in TI assembler.
The executable form of this program (tf.out) is loaded by the
program ladder.c onto the DSP platform for running on the
TMS320C30 DSP board.

Written by Eric lamarre 1992

TI C v.4.10 Compiler and Linker Directives

cl30 -o1 -s -al tf.c -z -cr -i c:\c30tools tf.cmd lsiboot.obj
i.obj fft_rl.obj ifft_rl.obj s1024.obj -l rts30.lib -o tf.out

also see the command file tf.cmd for other options
**************************************************/
```

```
asm (" .length 58");
asm (" .width 120");

#include <math.h>

#define FFT_SIZE 1024
#define FFT_POWER 10
#define OFFSET 4
#define MESH 20
#define NUM_LOC 1

extern long Comm0, Comm1;
extern float *Comm2, *Comm3, *Comm4, *Comm5;

void mean(int n, float *data, float *mmean);
void maxarray(int nn, int isign, float *data, float *max, int
*maxloc);
void interp(int nn, int locmax, int offset, int mesh_interp,
        float *data, float *rmax, float *rlocmax);
int fft_rl(int nn, int mm, float *data);
int ifft_rl(int nn, int mm, float *data);

long *PCproceedFlag, *DSPproceedFlag;
float data1[FFT_SIZE], data2[FFT_SIZE], s[FFT_SIZE];
float max_loc[NUM_LOC];

void main(void)
{
  long  i, j, k;
  int locmax;
  float max, rmax, rlocmax;
  float mean1, mean2;

/* INITIALIZATION OF POINTERS AND VARIABLES */

  PCproceedFlag = &Comm0;
  DSPproceedFlag = &Comm1;
  Comm4 = data1; Comm5 = data2; Comm3 = s; Comm2 =
max_loc;

/* INPUT DATA FROM PC */

  *PCproceedFlag = 1;
  beginning:
  while (*DSPproceedFlag == 0);
  *DSPproceedFlag = 0;

/* REMOVE MEAN OF THE TWO TIME-SERIES */

  mean(FFT_SIZE, data1, &mean1);
  mean(FFT_SIZE, data2, &mean2);

/* FFT OF THE TWO REAL DATA ARRAYS */

  fft_rl(FFT_SIZE, FFT_POWER, data1);
  fft_rl(FFT_SIZE, FFT_POWER, data2);

/* COMPUTATION OF CROSS-SPECTRAL DENSITY */

  s[0] = data1[0] * data2[0];
  s[FFT_SIZE/2] = data1[FFT_SIZE/2] * data2[FFT_SIZE/2];
  for (i=1; i<FFT_SIZE/2; i++)
  {
    j = FFT_SIZE - i;
    s[i] = data1[i] * data2[i] + data1[j] * data2[j];
  }
  for (i=1; i<FFT_SIZE/2; i++)
  {
    j = FFT_SIZE - i;
    s[j] = -data1[j] * data2[i] + data1[i] * data2[j];
  }

/* INVERSE TRANSFORM TO RECOVER CROSS-
CORRELATION */

  ifft_rl(FFT_SIZE, FFT_POWER, s);

/* EXTRACT PEAK OF CROSS-CORRELATION AND
INTERPOLATE AROUND THE PEAK */

  maxarray(FFT_SIZE, 1, s, &max, &locmax);
  interp(FFT_SIZE, locmax, OFFSET, MESH, s, &rmax,
&rlocmax);
  max_loc[0] = rlocmax;

/* OUTPUT RESULTS BACK TO PC */

  *PCproceedFlag = 1;
  while (*DSPproceedFlag == 0);
  *DSPproceedFlag = 0;

  goto beginning;
}
```

command file: tf.cmd

```
/*
TF.CMD
Memory map for LSI TMS320C30 System Board
for use with the C Compiler (When not using SPOX).

All memory is RAM.
*/
MEMORY
/* This maps memory SECTIONS to the board
hardware.*/
{
/*
EXTERNAL SRAM ON THE MAIN BOARD:
Locations 0 to C0h are reserved for interrupt vectors
and Debug Monitor usage.  Although you COULD start
using memory at C1h, this map starts at 100h -- to
allow for future Monitor expansion, and for ease of
adding hex addresses offsets.
*/
VECTS: origin=000000h  length=00000ch
   /* Interrupt vectors. */
BANK0: origin=000100h  length=00ff00h
   /*Std SRAM (0-wait). */
BANK1: origin=010000h  length=010000h
   /* SRAM upgrade option. */
BANK2: origin=020000h  length=010000h
   /* SRAM upgrade option. */
BANK3: origin=030000h  length=00f400h
   /* Std dual-access (1-wait). */
```

```
/*
Bank 3 is dual-access between the 'c30 and the PC. The
length shown is for the default 64Kx4 devices, but
16Kx4 can be used. In both cases, the top c00h
locations are reserved for Debug Monitor use. If you
will never use the debug monitor, your programs can
use this area.

CACHED DRAM MEMORY EXPANSION ON THE
DAUGHTER BOARD:
*/
EXPAND: origin=400000h  length=400000h
   /* One of various options. */
/*
ON-CHIP MEMORY:
*/
BLOCK0: origin=809800h  length=0000400h
BLOCK1: origin=809c00h  length=0000400h
}

SECTIONS
/*
Assigns program sections to the MEMORY statement,
above. The .data section, below, is not used by the
linker to link C compiler output files. It is used by the
linker when it is linking Assembler output files. The
section is included in this "map" file so that the same
map can be used to link files produced by EITHER the
Assembler or C Compiler (useful if you write some
functions in assembly language and happen to use the
.data section).
*/
{
   .text:        {}  >BANK0
   .bss:
   {
     _Comm0 = .;  . += 1;
     _Comm1 = .;  . += 1;
     _Comm2 = .;  . += 1;
     _Comm3 = .;  . += 1;
     _Comm4 = .;  . += 1;
     _Comm5 = .;  . += 1;
     _Comm6 = .;  . += 1;
     _Comm7 = .;  . += 1;
     _Comm8 = .;  . += 1;
     _Comm9 = .;  . += 1;
     _Comm10 = .;  . += 1;
     _Comm11 = .;  . += 1;
     _Comm12 = .;  . += 1;
     _Comm13 = .;  . += 1;
     _Comm14 = .;  . += 1;
     _Comm15 = .;  . += 1;

/* Define global address labels that can be used for
communication between the PC and the DSP programs.
These locations will each occupy one 32bits word
starting at zero offset from the beginning of the .bss
section.
If you need more locations, you could add more
"Comm" locations or you could create a "hole" in
memory here that you address using absolute pointers
(instead of these labels).
*/
   } >BANK3
   .data:        {}  >BANK3
   .cinit:       {}  >BANK3
   .stack:       {}  >BLOCK0

/* Forces Reset and Interrupt Vectors to absolute
locations: Your C source code should initialize these
locations using "ASM" in-line assembly macros (except
for location 00, which  is initialized in the
LSIBOOT.SRC startup file. */

    .int00  00h: {}
   /*Reset (Power-on or otherwise).*/
    .int01  01h: {}
   /* INT0 */
   .int02  02h: {}
   /* INT1 (A/D & D/A end of convert).*/
    .int03  03h: {}
   /* INT2 */
    .int04  04h: {}
   /* INT3 */
    .int05  05h: {}
   /* XINT0 */
    .int06  06h: {}
   /* RINT0 */
    .int07  07h: {}
   /* XINT1 */
    .int08  08h: {}
   /* RINT1 */
    .int09  09h: {}
   /* TINT0 */
    .int10  0ah: {}
   /* TINT1 */
    .int11  0bh: {}
   /* DINT */

}
```

functions: i.c

```
/****************************************************
i.c: Ensemble of functions used by profilt.c and tf.c
see these programs for compiler directives

Note: These functions run on Microsoft C V6.0 and TI C V.4.10

Functions written by Eric Lamarre, 1992
****************************************************/

void spline(float *x, float *y, int n, float yp1, float ypn,
        float *y2, float *u)
{
  int i,k;
```

```
 float p,qn,sig,un;

 if (yp1 > 0.99e30)
   y2[1]=u[1]=0.0;
   else {
   y2[1] = -0.5;
   u[1]=(3.0/(x[2]-x[1]))*((y[2]-y[1])/(x[2]-x[1])-yp1);
 }
 for (i=2;i<=n-1;i++) {
   sig=(x[i]-x[i-1])/(x[i+1]-x[i-1]);
   p=sig*y2[i-1]+2.0;
   y2[i]=(sig-1.0)/p;
   u[i]=(y[i+1]-y[i])/(x[i+1]-x[i]) - (y[i]-y[i-1])/(x[i]-x[i-1]);
   u[i]=(6.0*u[i]/(x[i+1]-x[i-1])-sig*u[i-1])/p;
 }
 if (ypn > 0.99e30)
   qn=un=0.0;
   else {
   qn=0.5;
   un=(3.0/(x[n]-x[n-1]))*(ypn-(y[n]-y[n-1])/(x[n]-x[n-1]));
 }
 y2[n]=(un-qn*u[n-1])/(qn*y2[n-1]+1.0);
 for (k=n-1;k>=1;k--)
   y2[k]=y2[k]*y2[k+1]+u[k];
}

void splint(float *xa, float *ya, float *y2a, int n, float x, float *y)
{
 int klo,khi,k;
 float h,b,a;

 klo=1;
 khi=n;
 while (khi-klo > 1) {
   k=(khi+klo) >> 1;
   if (xa[k] > x) khi=k;
   else klo=k;
 }
 h=xa[khi]-xa[klo];
 if (h == 0.0) exit(0);
/*if (h == 0.0) printf("Bad XA input to routine SPLINT"); */
 a=(xa[khi]-x)/h;
 b=(x-xa[klo])/h;
 *y=a*ya[klo]+b*ya[khi]+((a*a*a-a)*y2a[klo]+(b*b*b-
b)*y2a[khi])*(h*h)/6.0;
}

void maxarray(int nn, int isign, float *data, float *max, int
*locmax)
{
 int i, llocmax = 0;
 float mmax = 0.0;

 if (isign >= 0)
 {
   for (i = 0; i < nn; i++)
   {
     if (data[i] > mmax)
     { mmax = data[i]; llocmax = i; }
   }
 }
 else
 {
   for (i = 0; i < nn; i++)
   {
     if (data[i] < mmax)
     { mmax = data[i]; llocmax = i; }
   }
 }
 *max = mmax;
 *locmax = llocmax;
}

void interp(int nn, int locmax, int offset, int mesh_interp,
          float *data, float *rmax, float *rlocmax)
{
 int i, j;
 float sx[9], rtime[9], y2[9], u[9];
 float yp1, ypn, interp;
 float mesh, deltat, max;

/* EXTRACT PARTIAL ARRAY AROUND THE PEAK FOR
SPLINE FIT */

 yp1 = 1.0e31; ypn = 1.0e31;
 for (j = locmax-offset; j < locmax+offset+1; j++)
 {
   sx[j-(locmax-offset)] = data[j];
   rtime[j-(locmax-offset)] = j;
 }

/* CALL SPLINE ROUTINE TO GET 2ND DERIVATIVES */

 spline(rtime-1,sx-1,2*offset+1,yp1,ypn,y2-1,u-1);

/* CALL SPLINT TO INTERPOLATE ON FINER MESH (20
TIMES SMALLER) */

 i = 0;
 max = 0.0;
 deltat = 1.0 / mesh_interp;
 splint(rtime-1,sx-1,y2-1,2*offset+1,rtime[offset-1],&interp);
 while(interp > max)
 {
   i++;
   max = interp;
   splint(rtime-1,sx-1,y2-1,2*offset+1,rtime[offset-
1]+i*deltat,&interp);
 }
 *rlocmax = rtime[offset-1] + (i-1)*deltat;
 *rmax = max;
}


void mean(int n, float *data, float *mmean)
{
 int i;
 float sum=0.0;

 for (i=0; i<n; i++) sum = sum + data[i];
 *mmean = sum / (float) n;
 for (i=0; i<n; i++) { data[i] = data[i] - *mmean; }
}
```

function: fft_rl.asm

```
*
* fft_rl - radix-2 real FFT written in TI assembler and callable as
* a C function.
```

```
*
* Program (without modifications) to be found in the book
* Digital Signal Processing Applications (vol 3) by Papamichalis
*
* Synopsis:
*   int fft_rl(N, M, data)
*   int N FFT size: N=2**M
*   int M Number of stages = log2(N)
*   float *data Array with input and output data
*
* Description:
* Generic function to do a radix-2 FFT computation on the
* 320C30. The data array is N-long, with only real data. The
* output is stored in the same locations with real and imaginary
* points R and I as follows:
* R(0), R(1),...., R(N/2), I(N/2-1),...., I(1)
*
* The program is based on the FORTRAN program in the paper
* by Sorensen et al., June 1987 issue of Trans. on ASSP.
* The computation is done in place, and the original data is
* destroyed. Bit reversal is implemented at the beginning of the
* function. If this is not necessary, this part can be commented
* out.
*
* The sine/cosine table for the twiddle factors is expected to be
* supplied during link time, and it should have the following
* format:
*
*   .global   _sine
*   .data
* _sine .float      value1   = sin(0*2*pi/N)
*   .float      value2   = sin(1*2*pi/N)
*   ......
*   .float      value(N/2)   = cos((N/4)*2*pi/N)
*
* The values value1 to value(N/4) are the first quarter of the sine
* period, and value(N/4+1) to value(N/2) are the first quarter of
* the cosine period.
*
* Stack structure upon the call:
* -FP(4)      data
* -FP(3)      M
* -FP(2)      N
* -FP(1)      return addr
* -FP(0)      old FP
*
* Registers used: R0, R1, R2, R3, R4, R5, AR0, AR1, AR2,
* AR4, AR5, IR0, IR1, RS, RE, RC
*
* AUTHOR: PANOS E. PAPAMICHALIS
*     TEXAS INSTRUMENTS, OCTOBER 13, 1987
*
*****************************************************
* Modified to be a C callable routine by:
* George E. Peo, Jr.
* Stow Computer
* 111 Old Bolton Road
* Stow, MA  01775
* (508) 897-6838
* August 26,1992
*****************************************************
*
FP .set  AR3

   .GLOBL _fft_rl         ; ENTRY POINT FOR EXECUTION
   .GLOBL _sine          ; ADDRESS OF SINE TABLE

   .BSS     FFTSIZ,1
   .BSS     LOGFFT,1
   .BSS     INPUT,1

   .TEXT

SINTAB    .word    _sine

; INITIALIZE C FUNCTION

_fft_rl: PUSH    FP         ; SAVE DEDICATED REGISTERS
   LDI SP,FP
   PUSH    R4
   PUSH    R5
   PUSH    AR4
   PUSH    AR5

   LDI *-FP(2),R0   ; MOVE ARGUMENTS TO LOCATIONS
MATCHING
   STI R0,@FFTSIZ        ; THE NAMES IN THE PROGRAM
   LDI *-FP(3),R0
   STI R0,@LOGFFT
   LDI *-FP(4),R0
   STI R0,@INPUT

; DO THE BIT-REVERSING AT THE BEGINNING

   LDI @FFTSIZ,RC        ; RC=N
   SUBI     1,RC            ; RC SHOULD BE ONE LESS THAN
DESIRED #
   LDI @FFTSIZ,IR0
   LSH      -1,IR0          ; IR0=HALF THE SIZE OF FFT=N/2
   SUBI     R0,R0           ; MOD1
   LDI R0,AR0       ; MOD1
   LDI R0,AR1       ; MOD1
   LDI @INPUT,IR1       ; MOD1
;  LDI @INPUT,AR0       ; MOD1
;  LDI @INPUT,AR1       ; MOD1

   RPTB    BITRV
   CMPI    AR1,AR0           ; XCHANGE LOCATIONS
ONLY
   BGE      CONT          ; IF AR0<AR1
   LDF      *+AR0(IR1),R0  ; MOD1
|| LDF      *+AR1(IR1),R1  ; MOD1
   STF R0,*+AR1(IR1)  ; MOD1
|| STF R1,*+AR0(IR1)  ; MOD1
;  LDF      *AR0,R0           ; MOD1
;|| LDF     *AR1,R1           ; MOD1
;  STF R0,*AR1          ; MOD1
;|| STF R1,*AR0         ; MOD1
CONT  NOP    *AR0++
BITRV NOP    *AR1++(IR0)B

;  LENGTH-TWO BUTTERFLIES

   LDI @INPUT,AR0    ; AR0 POINTS TO X(I)
   LDI IR0,RC        ; REPEAT N/2 TIMES
   SUBI     1,RC          ; RC SHOULD BE ONE LESS THAN
DESIRED #

   RPTB   BLK1
   ADDF   *+AR0,*AR0++,R0  ; R0=X(I)+X(I+1)
```

```
    SUBF    *AR0,*-AR0,R1  ; R1=X(I)-X(I+1)
BLK1  STFR0,*-AR0          ; X(I)=X(I)+X(I+1)
||  STFR1,*AR0++  ; X(I+1)=X(I)-X(I+1)

; FIRST PASS OF THE DO-20 LOOP (STAGE K=2 IN DO-10
LOOP)

    LDI @INPUT,AR0    ; AR0 POINTS TO X(I)
    LDI 2,IR0         ; IR0=2=N2
    LDI @FFTSIZ,RC
    LSH     -2,RC         ; REPEAT N/4 TIMES
    SUBI    1,RC          ; RC SHOULD BE ONE LESS THAN
DESIRED #

    RPTB    BLK2
    ADDF    *+AR0(IR0),*AR0++(IR0),R0      ; R0 =
X(I)+X(I+2)
    SUBF    *AR0,*-AR0(IR0),R1 ; R1=X(I)-X(I+2)
    NEGF    *+AR0,R0          ; R0=-X(I+3)
||  STFR0,*-AR0(IR0)   ; X(I)=X(I)+X(I+2)
BLK2  STFR1,*AR0++(IR0) ; X(I+2)=X(I)-X(I+2)
||  STFR0,*+AR0          ; X(I+3)=-X(I+3)

; MAIN LOOP (FFT STAGES)

    LDI @FFTSIZ,IR0
    LSH     -2,IR0          ; IR0=INDEX FOR E
    LDI 3,R5            ; R5 HOLDS THE CURRENT STAGE
NUMBER
    LDI 1,R4            ; R4=N4
    LDI 2,R3            ; R3=N2
LOOP  LSH     -1,IR0          ; E=E/2
    LSH     1,R4            ; N4=2*N4
    LSH     1,R3            ; N2=2*N2

;   INNER LOOP (DO-20 LOOP IN THE PROGRAM)

    LDI @INPUT,AR5    ; AR5 POINTS TO X(I)
INLOP LDI IR0,AR0
    LDP     SINTAB          ; GEP SINTAB has a different DP
than FFTSIZ, etc
    ADDI    @SINTAB,AR0  ; AR0 POINTS TO SIN/COS
TABLE
    LDP     FFTSIZ        ; GEP restore DP
    LDI R4,IR1        ; IR1=N4

    LDI AR5,AR1
    ADDI    1,AR1         ; AR1 POINTS TO X(I1)=X(I+J)
    LDI AR1,AR3
    ADDI    R3,AR3        ; AR3 POINTS TO X(I3)=X(I+J+N2)
    LDI AR3,AR2
    SUBI    2,AR2         ; AR2 POINTS TO X(I2)=X(I-J+N2)
    ADDI    R3,AR2,AR4      ; AR4 POINTS TO X(I4)=X(I-
J+N1)

    LDF     *AR5++(IR1),R0; R0=X(I)
    ADDF    *+AR5(IR1),R0,R1    ; R1=X(I)+X(I+N2)
    SUBF    R0,*++AR5(IR1),R0  ; R0=-X(I)+X(I+N2)
||  STFR1,*-AR5(IR1)    ; X(I)=X(I)+X(I+N2)
    NEGF    R0        ; R0=X(I)-X(I+N2)
    NEGF    *++AR5(IR1),R1 ; R1=-X(I+N4+N2)
||  STFR0,*AR5            ; X(I+N2)=X(I)-X(I+N2)
    STFR1,*AR5            ; X(I+N4+N2)=-X(I+N4+N2)
```

```
; INNERMOST LOOP

    LDI @FFTSIZ,IR1
    LSH     -2,IR1          ; IR1=SEPARATION BETWEEN
SIN/COS TBLS
    LDI R4,RC
    SUBI    2,RC            ; REPEAT N4-1 TIMES

    RPTB    BLK3
    MPYF    *AR3,*+AR0(IR1),R0   ; R0=X(I3)*COS
    MPYF    *AR4,*AR0,R1   ; R1=X(I4)*SIN
    MPYF    *AR4,*+AR0(IR1),R1   ; R1=X(I4)*COS
||  ADDF    R0,R1,R2          ; R2 = X(I3)*COS+X(I4)*SIN
    MPYF    *AR3,*AR0++(IR0),R0 ; R0=X(I3)*SIN
    SUBF    R0,R1,R0          ; R0=-X(I3)*SIN+X(I4)*COS !!!
    SUBF    *AR2,R0,R1 ; R1=-X(I2)+R0 !!!
    ADDF    *AR2,R0,R1 ; R1=X(I2)+R0 !!!
||  STFR1,*AR3++  ; X(I3)=-X(I2)+R0 !!!
    ADDF    *AR1,R2,R1 ; R1=X(I1)+R2
||  STFR1,*AR4--   ; X(I4)=X(I2)+R0 !!!
    SUBF    R2,*AR1,R1 ; R1=X(I1)-R2
||  STFR1,*AR1++ ; X(I1)=X(I1)+R2
BLK3  STFR1,*AR2--   ; X(I2)=X(I1)-R2

    SUBI    @INPUT,AR5
;   ADDI    R3,AR5       ; AR5=I+N1  GEP
    ADDI    R4,AR5       ; AR5=I+N1  GEP
    CMPI    @FFTSIZ,AR5
;   BLED    INLOP       ; LOOP BACK TO THE INNER
LOOP
    BLTD    INLOP       ; GEP
    ADDI    @INPUT,AR5
    NOP
    NOP

    ADDI    1,R5
    CMPI    @LOGFFT,R5
    BLE     LOOP

; RESTORE THE REGISTER VALUES AND RETURN

    POP     AR5
    POP     AR4
    POP     R5
    POP     R4
    POP     FP
    RETS
```

function: ifft_rl.asm

```
; Function ifft_rl.asm written in TI assembler
; modified to work as C callable function
;
; Program to be found in the book
; Digital Signal Processing Applications (vol 3) by Papamichalis
;
; GENERIC PROGRAM TO DO A RADIX-2 REAL INVERSE
; FFT COMPUTATION ON 320C30.
;
; THE (REAL) DATA RESIDE IN INTERNAL MEMORY.
; THE COMPUTATION IS DONE IN-PLACE.  THE
; BIT-REVERSAL IS DONE AT THE BEGINNING OF
; THE PROGRAM.  THE INPUT DATA ARE STORED IN
; THE FOLLOWING ORDER:
```

```
; RE(0), RE(1),..., RE(N/2), IM(N/2-1),..., IM(1)
;
; THE TWIDLE FACTORS ARE SUPPLIED IN A TABLE PUT
; IN A .DATA SECTION.  THIS DATA IS INCLUDED IN A
; SEPARATE FILE TO PRESERVE THE GENERIC NATURE
; OF THE PROGRAM.  FOR THE SAME PURPOSE, THE SIZE
; OF THE FFT N AND LOG2(N) ARE DEFINED IN A .GLOBL
; DIRECTIVE AND SPECIFIED DURING LINKING.
; THE LENGTH OF THE TABLE IS N/4 + N/4 = N/2.
;
; AUTHOR: PANOS PAPAMICHALIS, DECEMBER 21, 1988
;
;
; Stack structure upon the call:
; -FP(4)data
; -FP(3)M
; -FP(2)N
; -FP(1)return addr
; -FP(0)      old FP
;
; Registers used: R0, R1, R2, R3, R4, R5, AR0, AR1, AR2, AR4,
 AR5, IR0, IR1, RS, RE, RC
;
*****************************************************
* Adapted to be a C callable routine by:
*   George E. Peo, Jr.
*   Stow Computer
*   111 Old Bolton Road
*   Stow, MA  01775
*   (508) 897-6838
*   August 26,1992
*****************************************************
;
FP .set  AR3

    .GLOBL_ifft_rl        ; ENTRY POINT FOR EXECUTION
    .GLOBL_sine           ; ADDRESS OF SINE TABLE

    .BSS     FFTSIZ,1
    .BSS     LOGFFT,1
    .BSS     INPUT,1

    .TEXT

SINTAB    .word    _sine

;  INITIALIZE C FUNCTION

_ifft_rl:PUSH   FP        ; SAVE DEDICATED REGISTERS
    LDI SP,FP
    PUSH   R4
    PUSH   R5
    PUSH   AR4
    PUSH   AR5

    LDI *-FP(2),R0   ; MOVE ARGUMENTS TO LOCATIONS
MATCHING
    STI R0,@FFTSIZ        ; THE NAMES IN THE PROGRAM
    LDI *-FP(3),R0
    STI R0,@LOGFFT
    LDI *-FP(4),R0
    STI R0,@INPUT

; MAIN LOOP (FFT STAGES)
```

```
    LDI 1,IR0          ; IR0=INDEX FOR E
    LDI 3,R5           ; R5 HOLDS THE CURRENT STAGE
NUMBER
    LDI @FFTSIZ,R3
    LSH     -1,R3          ; R3=N1/2=N2
    LDI @FFTSIZ,R4
    LSH     -2,R4          ; R4=N1/4=N4

;  INNER LOOP

LOOP  LDI @INPUT,AR5     ; AR5 POINTS TO X(I)
    LDI IR0,AR0
INLOP LDP     SINTAB           ; GEP SINTAB has a different
DP than FFTSIZ, etc
    ADDI    @SINTAB,AR0  ; AR0 POINTS TO SIN/COS
TABLE
    LDP     FFTSIZ       ; GEP restore DP

    LDI R4,IR1         ; IR1=N4
    LDI AR5,AR1
    ADDI    1,AR1          ; AR1 POINTS TO X(I1)=X(I+J)
    LDI AR1,AR3
    ADDI    R3,AR3         ; AR3 POINTS TO X(I3)=X(I+J+N2)
    LDI AR3,AR2
    SUBI    2,AR2          ; AR2 POINTS TO X(I2)=X(I-J+N2)
    ADDI    R3,AR2,AR4      ; AR4 POINTS TO X(I4)=X(I-
J+N1)

    NOP    *++AR5(IR1)      ; POINT TO X(I+N4)
    ADDF    *-AR5(IR1),*+AR5(IR1),R0
    SUBF    *+AR5(IR1),*-AR5(IR1),R1
    STF R0,*-AR5(IR1)   ; X(I)=X(I)+X(I+N2)
    STF R1,*++AR5(IR1); X(I+N2)=X(I)-X(I+N2)
||  LDF     *AR5,R0
    MPYF    2.0,R0
    STF R0,*-AR5(IR1)   ; X(I+N4) = 2*X(I+N4)
||  LDF     *++AR5(IR1),R1
    MPYF    -2.0,R1
    STF R1,*AR5++(IR1); X(I+N4+N2)=-X(I+N4+N2)*2


; INNERMOST LOOP

    LDI @FFTSIZ,IR1
    LSH     -2,IR1          ; IR1=SEPARATION BETWEEN
SIN/COS TBLS
    LDI R4,RC
    SUBI    2,RC            ; REPEAT N4-1 TIMES

    RPTB    BLK3
    SUBF    *AR2,*AR1,R1   ; R1=T1=X(I1)-X(I2)
    ADDF    *AR2,*AR1,R0
    MPYF    R1,*+AR0(IR1),R0   ; R0=T1*COS
||  STF R0,*AR1++ ; X(I1)=X(I1)+X(I2)
    ADDF    *AR3,*AR4,R2   ; R2=T2=X(I3)+X(I4)
    SUBF    *AR3,*AR4,R6
    MPYF    R2,*AR0,R6 ; R6=T2*SIN
||  STF R6,*AR2--   ; X(I2)=X(I4)-X(I3)
    SUBF    R6,R0
    MPYF    R2,*+AR0(IR1),R6   ; R6=T2*COS
||  STF R0,*AR3++ ; X(I3)=T1*COS-T2*SIN
    MPYF    R1,*AR0++(IR0),R0  ; R0=T1*SIN
    ADDF    R6,R0
BLK3  STF R0,*AR4--   ; X(I4) = T1*SIN+T2*COS
```

```
   SUBI    @INPUT,AR5
   CMPI    @FFTSIZ,AR5
   BLTD    INLOP        ; LOOP BACK TO THE INNER
LOOP
   ADDI    @INPUT,AR5
   LDI IR0,AR0
   NOP                  ; GEP

   ADDI    1,R5
   CMPI    @LOGFFT,R5
   BLED    LOOP
   LSH     1,IR0        ; E=E*2
   LSH     -1,R4        ; N4=N4/2
   LSH     -1,R3        ; N2=N2/2

; LAST PASS OF THE MAIN LOOP

   LDI @INPUT,AR0    ; AR0 POINTS TO X(I)
   LDI 2,IR0         ; IR0=2=N2
   LDI @FFTSIZ,RC
   LSH     -2,RC        ; REPEAT N/4 TIMES
   SUBI    1,RC         ; RC SHOULD BE ONE LESS THAN
DESIRED #

   LDF     *+AR0(IR0),R0  ; R0=X(I+2)
   RPTB    BLK2
   ADDF    R0,*AR0++(IR0),R1  ; R1=X(I)+X(I+2)
   SUBF    R0,*-AR0(IR0),R1   ; R1=X(I)-X(I+2)
|| STFR1,*-AR0(IR0)   ; X(I)=X(I)+X(I+2)
   STFR1,*AR0++ ; X(I+2)=X(I)-X(I+2)
|| LDF     *-AR0,R1
   MPYF  2.0,R1         ; R1=2.0*X(I+1)
   STFR1,*-AR0(IR0)   ; X(I+1)=2.0*X(I+1)
|| LDF     *AR0++,R1
   MPYF  -2.0,R1        ; R1=-2.0*X(I+3)
BLK2  STFR1,*-AR0          ; X(I+3)=-2.0*X(I+3)
|| LDF     *+AR0(IR0),R0  ; R0=X(I+4+2)

;  LENGTH-TWO BUTTERFLIES

   LDI @INPUT,AR0    ; AR0 POINTS TO X(I)
   LDI @FFTSIZ,RC
   LSH     -1,RC        ; REPEAT N/2 TIMES
   SUBI    1,RC         ; RC SHOULD BE ONE LESS THAN
DESIRED #

   RPTB    BLK1
   ADDF    *+AR0,*AR0++,R0   ; R0=X(I)+X(I+1)
   SUBF    *AR0,*-AR0,R1  ; R1=X(I)-X(I+1)
BLK1  STFR0,*-AR0         ; X(I)=X(I)+X(I+1)
|| STFR1,*AR0++ ; X(I+1)=X(I)-X(I+1)

; DO THE BIT-REVERSING AT THE END

   LDI @FFTSIZ,RC     ; RC=N
   SUBI    1,RC         ; RC SHOULD BE ONE LESS THAN
DESIRED #
   LDI @FFTSIZ,IR0
   LSH     -1,IR0       ; IR0=HALF THE SIZE OF FFT=N/2
   SUBI    R0,R0        ; MOD1
   LDI R0,AR0      ; MOD1
   LDI R0,AR1      ; MOD1
   LDI @INPUT,IR1     ; MOD1

   RPTB    BITRV
```

```
   CMPI    AR1,AR0         ; XCHANGE LOCATIONS
ONLY
   BGE     CONT         ;IF AR0 < AR1
   LDF     *+AR0(IR1),R0  ; MOD1
|| LDF     *+AR1(IR1),R1  ; MOD1
   STFR0,*+AR1(IR1)  ; MOD1
|| STFR1,*+AR0(IR1)  ; MOD1
CONT  NOP     *AR0++
BITRV NOP     *AR1++(IR0)B

; RESTORE THE REGISTER VALUES AND RETURN

   POP     AR5
   POP     AR4
   POP     R5
   POP     R4
   POP     FP
   RETS
```

input file: s1024.asm

```
*
* File to be linked with the source code for a
* 1024-point real FFT
*
* By Eric lamarre, 1992.
* To be used with FFT_RL.ASM and IFFT_RL.ASM
*
* Note, this table was shorten in order to save space.  The reader
* should be able to reconstruct the table with the following info.
*
* The values of the first N/4 floats are the first quarter of the
* period (N=1024).
*     .float  value1   => sin(0*2*pi/N)
*     .float  value2   => sin(1*2*pi/N)
*     ......
*     .float  valueN/4 => sin(N/4*2*pi/N)
*
* The next N/4 values are the first quarter of the cosine period
*     .float  valueN/4+1   => cos(0*2*pi/N)
*     .float  valueN/4+2   => cos(1*2*pi/N)
*     ......
*     .float  valueN/2     => cos(N/4*2*pi/N)
*
* Here we only give the first 10 values, this table should be
* completed up to N=1024 values.
*


   .globl   _sine
   .globl   N
   .globl   M

N  .set  1024
M  .set  10

   .data

_sine
   .float  0.000000
   .float  0.006136
   .float  0.012272
   .float  0.018407
   .float  0.024541
   .float  0.030675
```

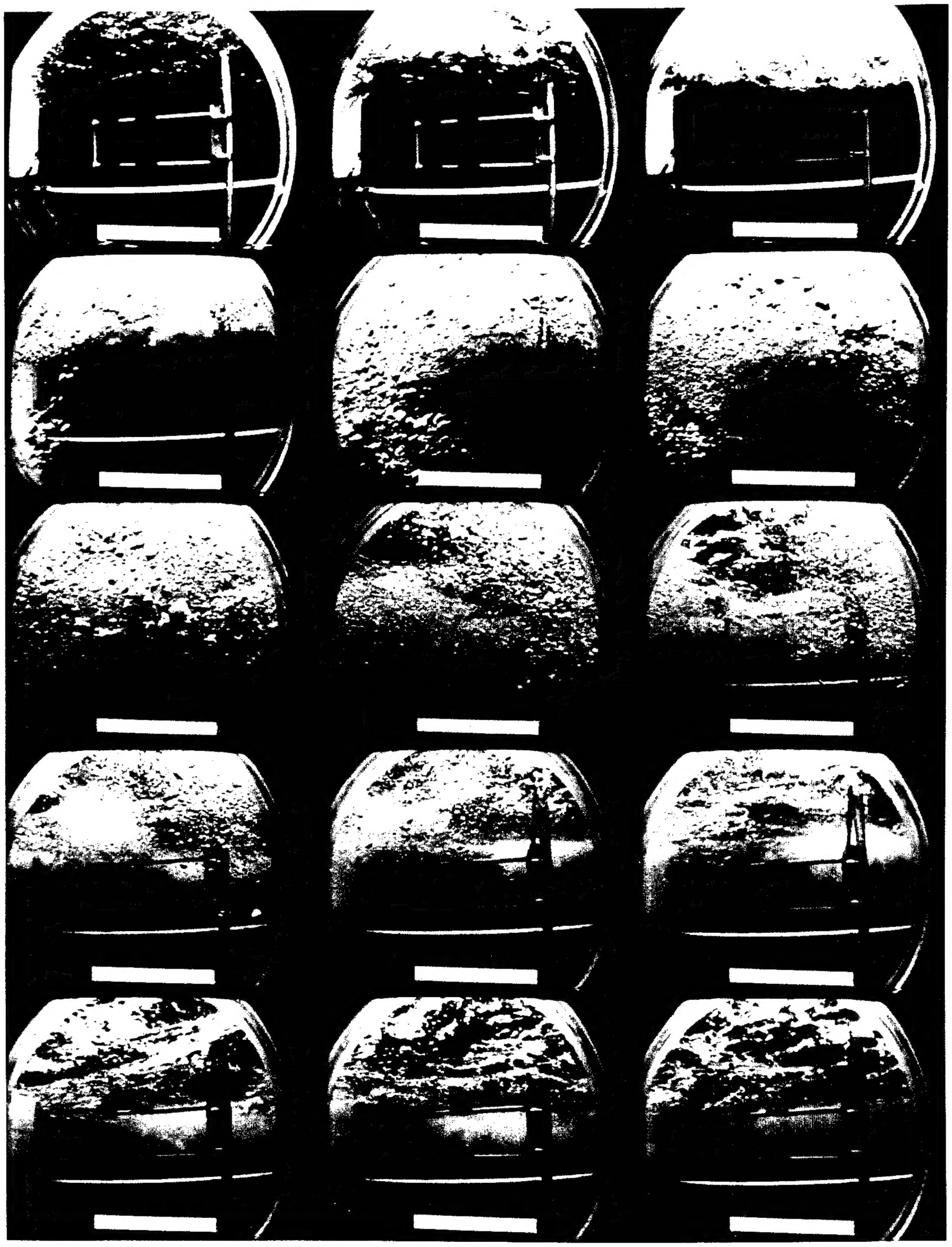
```
    .float  0.036807
    .float  0.042938
    .float  0.049068
    .float  0.055195
*   to be continued up to N
```

program: profilt.c

```
/***************************************************
File : profilt.c

Contains : Program is written in Microsoft C v6.0

Program to band-pass filter pulses by convoluting signal with
filter coefficients designed in Matlab.  The resoluting pulses are
cross-correlated and the time-delay and the attenuation is
computed.

First written by: Eric Lamarre (10-14-92)

COMPILING AND LINKING DIRECTIVES:
profilt.exe: profilt.obj lm30app.obj lmcload.obj i_pc.obj
LINK $**, $@, NUL, llibc7+llibfor7+smsclfft /NOD /NOE;

profilt.obj: profilt.c
CL /c /FPi87 /AL /G2 /Fs profilt.c
i_pc.obj: i_pc.c
CL /c /FPi87 /AL /G2 /Fs i_pc.c
***************************************************/

#include <stdio.h>
#include <string.h>
#include <conio.h>
#include <process.h>
#include <stdlib.h>
#include <math.h>
#include <dos.h>

#include "c:\c30tools\tms30.h"

#define DSP_BOARDADR 0x290
#define DSP_PROGRAM_NAME "tf.out"
#define NUM_LOC 5
#define COMM0 0x30000
#define PCPROCEEDFLAG COMM0 + 0
#define DSPPROCEEDFLAG COMM0 + 1
#define MAX_LOCADR COMM0 + 2
#define ANSADR COMM0 + 3
#define DATAR1ADR COMM0 + 4
#define DATAR2ADR COMM0 + 5
#define MAXCOUNT 2000000

#define N 1024
#define REPETITION 160
#define NUM_MODULE 1

#define MESH 20
#define OFFSET 4
#define SRATE 500000.0

#define EXP_NAME "sd"
#define CONFIG_NAME "1"
#define RAW_DATA_FILE_EXT ".raw"
#define RAW_DATA_DIR "d:\\data\\"
#define PRO_DATA_FILE_EXT ".pro"
#define AMP_DATA_FILE_EXT ".amp"
#define PRO_FILE_DIR "c:\\freq\\"

void EoLine(FILE *fp) { while(fgetc(fp) != '\n'); }
void initialize_dsp(void);
void mean(int n, float *data, float *mmean);
void maxarray(int nn, int isign, float *data, float *max, int
*maxloc);
void interp(int nn, int locmax, int offset, int mesh_interp,
        float *data, float *rmax, float *rlocmax);
void win_ts(int n, int winsize, float *window, float *data, int
locmax);
void hanning(int n, float *window);

int data1[N], data2[N];
float datar1[N], datar2[N], max_loc[NUM_LOC], s[N];
float
nmax_corr_loc[REPETITION*NUM_MODULE][NUM_LOC];
float
nmax_corr_amp[REPETITION*NUM_MODULE][NUM_LOC]
;
float filt[N], ffilt[2*N+2], d1[2*N+2], d2[2*N+2];

void main(int argc, char *argv[])
{
  FILE *fp, *fpp;
  int st, rep, type, nfilt, exp;
  int r, m, mm, i, j, k, f, numstart, numend, beg, end, flag;
  int locmax, locmin;
  int ddown[NUM_LOC], dup[NUM_LOC];
  int loctransm, halfper, reclocmax, reclocmin, win;
  float maxtransm, recmax, recmin;
  float max, min, rmax, rlocmax;
  float freq, mmean, *window;
  unsigned long datar1adrloc, datar2adrloc, ansadrloc,
max_locadrloc;
  char option[1], filename[30], pro_file[30], *padded_num,
namecopy[80];

/* GET COPROCESSOR TYPE */

  getndp(&type);

/* EXTRACT COMMAND LINE OPTION */

  if (argc < 6) { printf("Need at least three arguments!\n");
exit(st); }
  else
  {
    strcpy(option,argv[1]);
    numstart = (int) atof(argv[2]);
    numend = (int) atof(argv[3]);
    rep = (int) atof(argv[4]);
    freq = atof(argv[5]);
  }
  printf("freq = %f\n",freq);

/* READ FILTER COEFFICIENTS */

  if (freq == 5000.0)
  {
    nfilt = 513;
    if ((fp = fopen("filt5k.cof","rt")) == NULL)
    { printf("Cannot open file filt5k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 10000.0)
 {
  nfilt = 257;
  if ((fp = fopen("filt10k.cof","rt")) == NULL)
  { printf("Cannot open file filt10k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 15000.0)
 {
  nfilt = 257;
  if ((fp = fopen("filt15k.cof","rt")) == NULL)
  { printf("Cannot open file filt15k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 20000.0)
 {
  nfilt = 129;
  if ((fp = fopen("filt20k.cof","rt")) == NULL)
  { printf("Cannot open file filt20k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 25000.0)
 {
  nfilt = 129;
  if ((fp = fopen("filt25k.cof","rt")) == NULL)
  { printf("Cannot open file filt25k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 30000.0)
 {
  nfilt = 129;
  if ((fp = fopen("filt30k.cof","rt")) == NULL)
  { printf("Cannot open file filt30k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 35000.0)
 {
  nfilt = 129;
  if ((fp = fopen("filt35k.cof","rt")) == NULL)
  { printf("Cannot open file filt35k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }
 if (freq == 40000.0)
 {
  nfilt = 65;
  if ((fp = fopen("filt40k.cof","rt")) == NULL)
  { printf("Cannot open file filt40k.cof\n"); exit(st); }
  for (i=0; i<nfilt; i++) fscanf(fp,"%f\n",&filt[i]);
  close(fp);
 }

/* DEFINE A WINDOW ARRAY */

  win = 3.0 * (int) (SRATE * (1.0/(freq)));
  if ((win%2) != 0) win=win+1;
  window = (float *) calloc(win,sizeof(float));

/* INITIALIZATION OF DSP */

  initialize_dsp();
   ansadrloc = Get32Bit(ANSADR,DUAL);
   datar1adrloc = Get32Bit(DATAR1ADR,DUAL);
   datar2adrloc = Get32Bit(DATAR2ADR,DUAL);
   max_locadrloc = Get32Bit(MAX_LOCADR,DUAL);

/* START MAIN LOOP TO OPEN MANY FILES STARTING
AT FILENUM numstart AND */
/* ENDING AT FILENUM numend */

  for (f=numstart; f<numend+1; f++)
  {
  sprintf(padded_num,"%04d",f);
  strcpy(filename,RAW_DATA_DIR);
  strcat(filename,EXP_NAME);
  strcat(filename,CONFIG_NAME);
  strcat(filename,option);
  strcat(filename,padded_num);
  strcat(filename,RAW_DATA_FILE_EXT);
  printf("%s\n",filename);
  if ((fp = fopen(filename,"rb")) == NULL)
  { printf("Cannot open file %s\n",filename); exit(st); }

/* START MAIN REPETITION LOOP */

  printf("%s\n",filename);
  for (r=0; r<rep; r++)
  {
    printf("r = %d: ",r);
    for (m=0; m<NUM_MODULE; m++)
    {
      printf(" %d",m);

/* READ DATA TO DISK */

      fread(data1, sizeof(int), N, fp);
      fread(data2, sizeof(int), N, fp);
      for (i=0; i<N; i++) { datar1[i]=data1[i]; datar2[i]=data2[i];
}
      for (i=0; i<6-NUM_MODULE; i++)
      {
        fread(data1, sizeof(int), N, fp);
        fread(data2, sizeof(int), N, fp);
      }

/* REMOVE MEAN AND BANDPASS FILTER THE DATA */

      mean(N, datar1, &mmean);
      mean(N, datar2, &mmean);
      for (i=0; i<2*N; i++) { d1[i]=0.0; d2[i]=0.0; ffilt[i]=0.0;}
      for (i=0; i<N; i++) { d1[i]=datar1[i]; d2[i]=datar2[i]; }
      for (i=0; i<nfilt; i++) ffilt[i]=filt[i];
      exp = 11;
      conv(ffilt,d1,&exp);
      conv2(ffilt,d2,&exp);

/* WRITE FIRST RECORD TO DISK */

      if (r == 0 && m == 0)
      {
       if ((fpp = fopen("profilt.dat","wt")) == NULL)
       { printf("Cannot open file profilt.dat\n"); exit(st); }
```

```
        for (i=0;i<N;i++)
          fprintf(fpp,"%7.2f %7.2f %7.2f
%7.2f\n",datar1[i],datar2[i],
d1[i+(nfilt+1)/2],d2[i+(nfilt+1)/2]);
        }
        close(fpp);

/* WINDOW THE BANDPASSED TRANSMIT TIME-SERIES
AROUND ITS PEAK */

        for (i=0; i<N; i++)
        { datar1[i]=d1[i+(nfilt+1)/2]; datar2[i]=d2[i+(nfilt+1)/2]; }
        hanning(win, window);
        maxarray(N, 1, datar1, &maxtransm, &loctransm);
        win_ts(N, win, window, datar1, loctransm);

/* COMPUTE CROSS-CORRELATION (NEW WAY) */

        WrBlkFlt(datar1adrloc,DUAL,N,datar1);
        WrBlkFlt(datar2adrloc,DUAL,N,datar2);
        Put32Bit(DSPPROCEEDFLAG,DUAL,0x1L);

        while(Get32Bit(PCPROCEEDFLAG,DUAL) != 0x1L);
        Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
        RdBlkFlt(max_locadrloc,DUAL,NUM_LOC,max_loc);
        RdBlkFlt(ansadrloc,DUAL,N,s);
        Put32Bit(DSPPROCEEDFLAG,DUAL,0x1L);

/* EXTRACT PEAK OF CROSS-CORRELATION AND
INTERPOLATE AROUND THE PEAK */

        for (i=0; i<NUM_LOC; i++) { ddown[i]=-1; dup[i]=-1; }
        for (k = 0; k < NUM_LOC; k++)
        {
          max = 0.0;
          for (i = 0; i < N; i++)
          {
            for (j = 0; j < NUM_LOC; j++)
            {
              if (i >= ddown[j] && i <= dup[j]) { flag = 1; }
            }
            if (flag == 0)
            {
              if (s[i] > max) { max = s[i]; locmax = i; }
            }
            flag = 0;
          }
          if (locmax >= N-OFFSET) locmax = N-OFFSET-1;
          if (locmax <= OFFSET) locmax = OFFSET+1;
          ddown[k] = locmax - (int)(SRATE * (1.0/(2*freq)));
          dup[k] = locmax + (int)(SRATE * (1.0/(2*freq)));
          interp(N, locmax, OFFSET, MESH, s, &rmax,
&rlocmax);
          nmax_corr_loc[r*NUM_MODULE+m][k] = rlocmax;

/* MEASURE ATTENUATION BETWEEN MAIN PEAK AND
TROUGH */

          halfper = (int) (SRATE * (1.0/(2*freq)));
          maxarray(N, 1, datar1, &maxtransm, &loctransm);
          recmax = -2048.0;
          if (loctransm+locmax-halfper < 0) beg = 0;
          else beg = loctransm+locmax-halfper;
          if (loctransm+locmax+halfper > N) end = N;
          else end = loctransm+locmax+halfper;
          for (i=beg; i<end; i++)
          {
            if (datar2[i] > recmax) { recmax = datar2[i]; reclocmax
= i; }
          }
          recmin = 2048.0;
          if (loctransm+locmax-2*halfper < 0) beg = 0;
          else beg = loctransm+locmax-2*halfper;
          if (loctransm+locmax > N) end = N;
          else end = loctransm+locmax;
          for (i=beg; i<end; i++)
          {
            if (datar2[i] < recmin) { recmin = datar2[i]; reclocmin =
i; }
          }
          nmax_corr_amp[r*NUM_MODULE+m][k] = (recmax -
recmin)*10.0/2048.0;
        }
      }
      printf("\n");
    }
    printf("\n");

/* SAVE CROSS-CORRELATION RESULTS TO DISK */

    strcpy(pro_file,PRO_FILE_DIR);
    strcat(pro_file,EXP_NAME);
    strcat(pro_file,CONFIG_NAME);
    strcat(pro_file,option);
    strcat(pro_file,padded_num);
    strcat(pro_file,PRO_DATA_FILE_EXT);

    if ((fpp = fopen(pro_file,"wt")) == NULL)
    { printf("Cannot open file %s\n",pro_file); exit(st); }
    for (r=0; r<rep; r++)
    {
      for (m=0; m<NUM_MODULE; m++)
      {
        for (i=0; i<NUM_LOC; i++)
          fprintf(fpp,"%7.2f
",nmax_corr_loc[r*NUM_MODULE+m][i]);
        fprintf(fpp,"  ");
      }
      fprintf(fpp,"\n");
    }
    close(fpp);

/* SAVE AMPLITUDE INFORMATION */

    strcpy(pro_file,PRO_FILE_DIR);
    strcat(pro_file,EXP_NAME);
    strcat(pro_file,CONFIG_NAME);
    strcat(pro_file,option);
    strcat(pro_file,padded_num);
    strcat(pro_file,AMP_DATA_FILE_EXT);

    if ((fpp = fopen(pro_file,"wt")) == NULL)
    { printf("Cannot open file %s\n",pro_file); exit(st); }
    for (r=0; r<rep; r++)
    {
      for (m=0; m<NUM_MODULE; m++)
      {
        for (i=0; i<NUM_LOC; i++)
          fprintf(fpp,"%6.3f
",nmax_corr_amp[r*NUM_MODULE+m][i]);
```

```
      fprintf(fpp,"  ");
    }
    fprintf(fpp,"\n");
  }
  close(fpp);
  close(fp);
  } /* END OF LOOP WHERE WE PROCESS A NEW FILE
NAME */
} /* END OF MAIN */


void initialize_dsp(void)
{
  unsigned short loadstat;
  long i;

/* DOWNLOAD DSP CODE - CODE MUST BE IN SAME DIR
AS MAIN PROG. */

  SelectBoard(DSP_BOARDADR);
  loadstat = coffLoad(DSP_PROGRAM_NAME);
  if (loadstat != 0)
  {
    printf("\n\nError During Program Load.\n");
    printf("coffLoad() returned %x\n\n", loadstat);
    exit (0);
  }
  Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
  Put32Bit(DSPPROCEEDFLAG,DUAL,0x0L);
  Reset();

/* WAIT FOR DSP BOARD TO INITIALIZE */

  for (i=0; i<MAXCOUNT &&
(Get32Bit(PCPROCEEDFLAG,DUAL) != 0x1L); i++);
  if (i==MAXCOUNT) { printf("Timeout waiting for DSP.\n");
exit(0); }
  Put32Bit(PCPROCEEDFLAG,DUAL,0x0L);
}

void hanning(int n, float *window)
{
  int i;
  float pi = 3.141592654;
  for (i=0; i<n; i++) window[i] = 0.5 * (1.0 - cos(2.0 * pi * (i) /
(n-1)));
}

void win_ts(int n, int winsize, float *window, float *data, int
locmax)
/* ALWAYS USE AN ODD SIZE WINDOW */
{
  int i, down, up;

  down = locmax - (winsize-1)/2;
  up = locmax + (winsize-1)/2;
  for (i=0; i<down; i++) data[i] = 0.0;
  for (i=down; i<up; i++) data[i] = data[i] * window[i-down];
  for (i=up; i<n; i++) data[i] = 0.0;
}
```



50272-101

REPORT DOCUMENTATION PAGE	1. REPORT NO. WHOI-93-25	2.	3. Recipient's Accession No.
4. Title and Subtitle An Experimental Study of Air Entrainment by Breaking Waves			5. Report Date June 1993
			6.
7. Author(s) Eric Lamarre			8. Performing Organization Rept. No.
9. Performing Organization Name and Address Woods Hole Oceanographic Institution Woods Hole, Massachusetts 02543			10. Project/Task/Work Unit No. WHOI-93-25
			11. Contract(C) or Grant(G) No. (C) (G)
12. Sponsoring Organization Name and Address Funding was provided by the Office of Naval Research and the National Science Foundation through the Massachusetts Institute of Technology.			13. Type of Report & Period Covered Ph.D. Thesis
			14.

15. Supplementary Notes
This thesis should be cited as: Eric Lamarre, 1993.
An Experimental Study of Air Entrainment by Breaking Waves. Ph.D. Thesis. MIT/WHOI, WHOI-93-25.

16. Abstract (Limit: 200 words)

This work reports on a series of laboratory and field experiments on the measurement of air entrainment by breaking waves. The first part of this thesis addresses the measurement of high volumetric concentrations of air (0.3% to 100% void-fraction) found immediately beneath breaking waves. Instrumentation based on the change of electrical impedance of the bubbly mixture is developed. Laboratory and field measurements are conducted. Maps of the evolution of the void-fraction distribution for various size breaking waves yield several bulk characteristics of the air entrainment process. In particular, up to 95% of the initially entrained air volume is lost in the first wave period after breaking and up to 50% of the energy dissipated by breaking is found to be expended in entraining bubbles against their buoyancy.

The second part addresses the measurement of very low void-fractions. Instrumentation based on the propagation velocity of low-frequency acoustic pulses is developed. Simultaneous measurements of the sound-speed at several depths are conducted during two field experiments. Time-series of sound-speed and attenuation show dramatic fluctuations over time periods on the order of minutes or less which are attributed to bubble clouds. The time-averaged sound-speed profile is found to have lower velocities and to be shallower than previously reported.

17. Document Analysis a. Descriptors
breaking waves
air entrainment

b. Identifiers/Open-Ended Terms

c. COSATI Field/Group

18. Availability Statement Approved for publication; distribution unlimited.	19. Security Class (This Report) UNCLASSIFIED	21. No. of Pages 329
	20. Security Class (This Page)	22. Price

(See ANSI-Z39.18) *See Instructions on Reverse* OPTIONAL FORM 272 (4-77)
(Formerly NTIS-35)
Department of Commerce